Deep Sub-Region Network for Salient Object Detection

Liansheng Wang, *Member, IEEE*, Rongzhen Chen, Lei Zhu, *Member, IEEE*, Haoran Xie, *Senior Member, IEEE*, and Xiaomeng Li, *Member, IEEE*

***Abstract*— Saliency detection is a fundamental and challenging task in computer vision, which aims at distinguishing the most conspicuous objects or regions in an image. Existing deep-learning methods mainly rely on the entire image to learn the global context information for saliency detection, which loses the spatial relation and results in ambiguity in predicting saliency maps. In this paper, we propose a novel deep sub-region network (DSR-Net) equipped with a sequence of sub-region dilated blocks (*SRDB*) by aggregating multi-scale salient context information of multiple sub-regions, such that the global context information from the whole image and local contexts from sub-regions are fused together, making the saliency prediction more accurate. Our *SRDB* separates the input feature map at different layers of a convolutional neural network (CNN) into different sub-regions and then designs a parallel ASPP module to refine feature maps at each sub-region. Experiments on the five widely-used saliency benchmark datasets demonstrate that our network outperforms recent state-of-the-art saliency detectors quantitatively and qualitatively on all the benchmarks.**

***Index Terms*— Saliency detection, deep subregion learning, region dilated blocks, parallel atrous spatial pyramid pooling (ASPP) modules.**

I. Introduction

SALIENCY detection aims at highlighting the most visually distinctive objects or regions from an image [1]–[5]. Served as a pre-processing step, inferring salient objects plays an essential role in lots of computer vision applications, such as weakly supervised object detection [6], object recognition [7], image and video compression [8], [9], texture smoothing [10], and visual tracking [11], [12]. Saliency detection requires both an understanding of the whole image and an accurate identification of the details of salient regions. Hence, saliency detection is a challenging research problem [13]–[16].

Early works designed hand-crafted features and heuristic priors [17]–[19] to distinguish salient objects and non-salient backgrounds from the input image. However, these methods tend to fail in generating satisfactory results of saliency detection since their human-designed features lack high-level semantic information, which is required for inferring salient objects. To alleviate this problem, fully convolutional neural network (FCN) based methods have achieved remarkable saliency detection results by learning convolutional features at deep convolutional layers [20], [21] or integrating feature maps at multiple layers [22]–[24] of a convolutional neural network (CNN).

Recent CNN-based works detected salient objects by learning global contextual features [25], [26] for enlarging the convolutional receptive fields, or adding extra boundary information [27]–[30]. Although improving the saliency detection performance, these works fail in generating high-quality saliency detection results on complex scenes, since the global contextual features in these methods are directly learned from the whole input image, which loses the spatial relations and causes ambiguity for saliency detection. The global context information along with local contexts together has demonstrated to be more accurate and reliable by forming a more powerful feature representation in many works, including the classical bag-of-words based image classification [31], the spatial pyramid pooling for visual recognition [32], and the pyramid pooling module for semantic segmentation [33]. The success of these works is the starting point of our network for salient object detection.

In this paper, we propose a novel deep sub-region network (denoted as DSR-Net) to produce more accurate predictions for saliency detection by integrating saliency context features from sub-regions. Intuitively, image sub-regions have less non-salient details than the whole input image, and thus reduce the interference from non-salient objects, making the saliency prediction more accurate, especially for tiny salient objects (see Fig. 1). To do so, the DSR-Net develops a sub-region dilated block (*SRDB*) to refine the deep layers of a convolutional neural network (CNN) by learning saliency context information from multiple sub-regions with different receptive fields, and then combining the refined features from multiple deep CNN layers to generate the final prediction of our network. Experiments on the five widely-used benchmark datasets

Manuscript received October 2, 2019; revised January 26, 2020 and March 9, 2020; accepted April 10, 2020. Date of publication April 20, 2020; date of current version February 4, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61671399, in part by the National Natural Science Foundation of China under Grant 61902275, in part by the Fundamental Research Funds for the Central Universities under Grant 20720190012, in part by the Interdisciplinary Research Scheme of the Dean's Research Fund 2018-19 (FLASS/DRF/IDS-3) of The Education University of Hong Kong, and in part by the HKIBS Research Seed Fund 2019/20, Lingnan University, Hong Kong, under Grant 190-009. This article was recommended by Associate Editor C. Shen. *(Corresponding author: Lei Zhu.)*

Liansheng Wang and Rongzhen Chen are with the Department of Computer Science, School of Informatics, Xiamen University, Xiamen 361005, China.

Lei Zhu and Xiaomeng Li are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: lzhu@cse.cuhk.edu.hk).

Haoran Xie is with the Department of Computing and Decision Sciences, Lingnan University, Hong Kong.

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2020.2988768

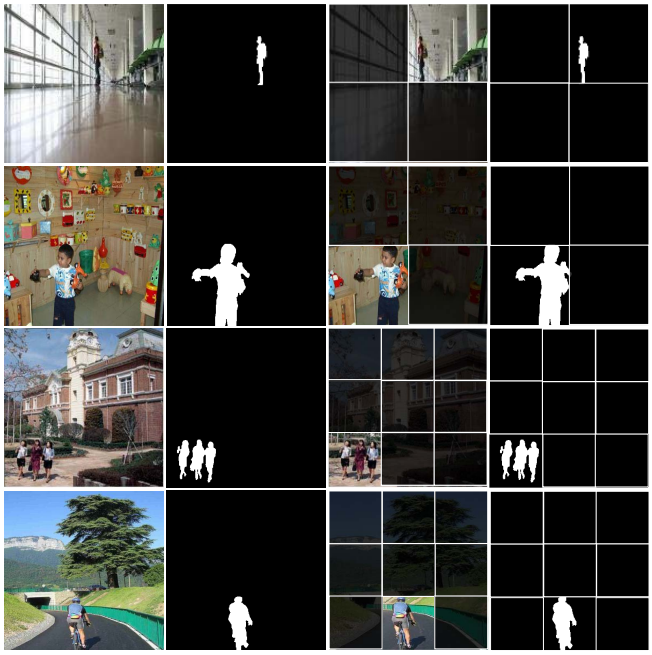


Fig. 1. Sub-region observation. The 1st and 2nd rows show the separation of the input image into four sub-regions (2×2), while the last two rows show the separation scheme to nine sub-regions (3×3). Apparently, the salient objects are more conspicuous in the sub-region due to alleviating non-salient interference.

demonstrate that our network boosts the saliency detection performance on all the five benchmarks, when compared to recent state-of-the-art methods.

Overall, our main contributions are summarized as follows.

- First, we design the sub-region dilated block (SRDB) to fuse the global contexts from the whole feature map and local contexts from local sub-regions together for learning more powerful features for salient object detection.
- Second, we develop a deep sub-region network equipped with sequences of SRDBs to progressively refine features at different CNN layers by learning context features from multiple sub-regions with different receptive fields. Such a sub-region based scheme is capable to learn more powerful discriminative features and has the potential to be adapted for other vision applications such as shadow detection, object detection, and semantic segmentation.
- Third, we evaluate the effectiveness of our method on five common benchmark datasets by comparing it against 26 state-of-the-art saliency detectors. Overall, our method consistently has the best performance of detecting salient objects on all the five benchmark datasets.

II. Related Work

In this section, we aim to review methods for salient object detection. Early attempts detected salient objects by exploiting human-designed visual attributes (*e.g*, contrast [18], [34], color [35], [36], texture [37], [38]) and some other low-level visual cues [39]. Unfortunately, these methods often fail to achieve convincing predictions since their hand-crafted priors from human observations do not always hold in diverse real-world photos.

Motivated by the remarkable results of CNN in diverse vision tasks, many methods [21], [40]–[43] have been proposed by designing different convolutional neural networks (CNNs) to detect salient objects in the past five years. It has also become apparent that accounting for the advanced semantic context brings more accurate detection of salient objects. Zhao *et al.* [21] utilized a multi-context deep learning framework to detect salient objects by taking both global and local context into account. Li *et al.* [40] proposed a multi-task CNN framework to train two tasks (saliency detection and object segmentation) for further boosting the feature representations for salient objects. Zhang *et al.* [43] combined the R-dropout technique with common convolutional operations to learn deep uncertain features for saliency detection. However, since these methods only leveraged the features from deep CNN layers to generate the final prediction, their results are rough and have a missing/ambiguous boundary.

In order to alleviate this issue, several works [22], [23], [44] integrated deep and shallow layer features to generate a more refined prediction maps by leveraging the complement information among different CNN layers. Li and Yu [45] utilized semantic properties of salient objects and visual contrasts among multi-scale feature maps for salient object detection. Hou *et al.* [22] utilized the short connections for aggregating multi-level features. Wang *et al.* [46] conducted the fixation prediction from a global/high-level view, built another CNN for salient object detection from lower layers, and combined them together for a better prediction of salient objects. Later, Deng *et al.* [24] and Chen *et al.* [47] both took advantage of efficient residual learning to progressively refine the saliency map predictions by integrating features at shallow and deep layers recurrently. In [26], an attention guided mechanism was designed to selectively integrate multi-level contextual information gradually. Starting from the same point, Zhang *et al.* [48] similarly developed a bi-directional message passing model to control the information flow during the multi-level features integration.

Very recently, many CNNs focused on extracting global context information [25], [49] to enlarge the convolutional receptive fields, or adding extra information (e.g., the image caption [50], and salient object boundaries [28]–[30], [50] for salient object detection. Wang *et al.* [49] proposed a recurrent module that integrates global contextual knowledge as time evolves, to locate salient objects more accurately. Liu *et al.* [25] generated the global/local attention contextual maps for each pixels in the feature maps, in order to selectively aggregating the global/local contextual information. On the other hand, Wu *et al.* [29] presented a saliency detection network by imposing the multi-task intertwined supervision from not only saliency detection, but also foreground contour detection and edge detection. Furthermore, Feng *et al.* [30] designed a boundary-aware network, which consists of a global perception module and a set of attentive feedback modules for saliency prediction, as well as a boundary-enhanced loss to assist the saliency detection on the object boundaries. Qin *et al.* [28] proposed a predict-refine network with a boundary-aware module and a hybrid fusing loss for salient object detection. Zhang *et al.* [50] formulate

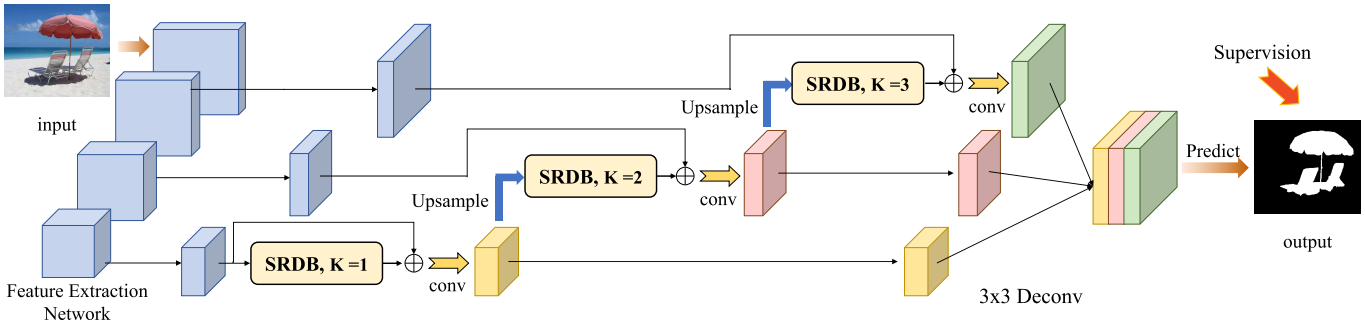


Fig. 2. The schematic illustration of the deep sub-region network equipped with SRDBs (see Figure 3 for the meaning of K). We empirically apply three SRDBs in our network by balancing the time performance and the saliency detection accuracy.

a network by leveraging image captions to further extract the semantic information to recognize salient objects. Although the detection quality still keeps improving on the benchmarks [20], [37], [38], [51]–[53], there exists a heavy loss of local spatial semantic information. However, existing saliency detection networks took the whole image as a consideration and determined the salient objects mainly relying on global semantic features, and thus failed to generate high-quality saliency maps for complex scenes or tiny objects. To further boost the saliency detection performance, we leverage the global context features along with local contexts from multiple sub-regions to learn a more powerful feature representation for salient object detection.

III. Our Approach

Fig. 2 shows the architecture of the developed deep sub-region network(DSR-Net), which takes the whole image as the input and predicts a saliency map as the output in an end-to-end manner. DSR-Net first utilizes a CNN as the feature extraction network to generate a set of feature maps with different spatial resolutions. Then, at each CNN layer, we design a sub-region dilated block (SRDB) to refine feature map at previous layer by fusing the global context features along with local contexts from sub-regions, and then merge the output feature map with the features at the current layer for the feature refinement. After that, we upsample the refined features at different CNN layers to the $1/4$ spatial size of the input image and concatenate them for predicting the saliency map, which is taken as the final output of our network.

The key idea of our DSR-Net is to design different SRBDs for learning more efficient semantic representations of salient objects by fusing the global context information from the whole feature map and local contexts from multiple sub-regions. The SRDB first separates the whole feature map into different sub-regions, and then designs a parallel atrous spatial pyramid pooling (ASPP) block to extract local context features from each sub-region, which are then merged with the input features to generate the output feature map.

In the following subsections, we will elaborate on the details of the sub-region dilated block (SRDB) in Section III-A and the parallel ASPP block in Section III-B. Section III-C summarizes the training and testing strategies of our network.

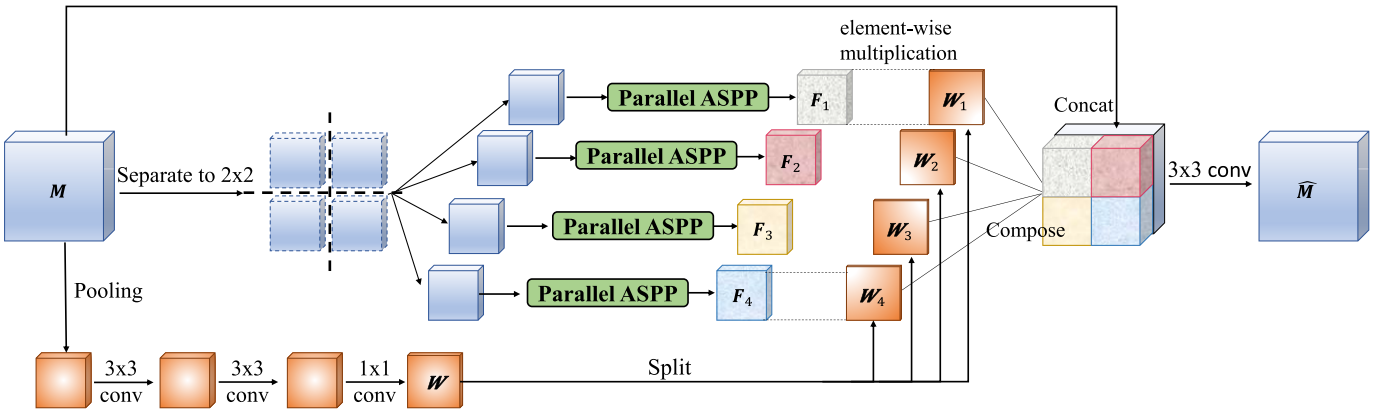
A. Sub-Region Dilated Block

Saliency detection is to seek the most conspicuous objects or regions in an input image. Due to complicated background scenarios in the input images, many non-salient background regions weaken visual representations of salient objects (especially for tiny targets (see Fig. 1)), and thus largely affect the image understanding for inferring saliency detection. Based on the observation that salient objects in sub-regions becomes more conspicuous, we develop a sub-region based convolutional neural network to boost the saliency detection, since there are less interference from non-salient backgrounds when using sub-region information for saliency detection. Our network has sub-region dilated blocks (SRDBs; see Fig. 3a) to refine feature maps at different CNN layers, and the SRDB divides the whole feature map into multiple sub-regions and then aggregates the context semantic information from each sub-region.

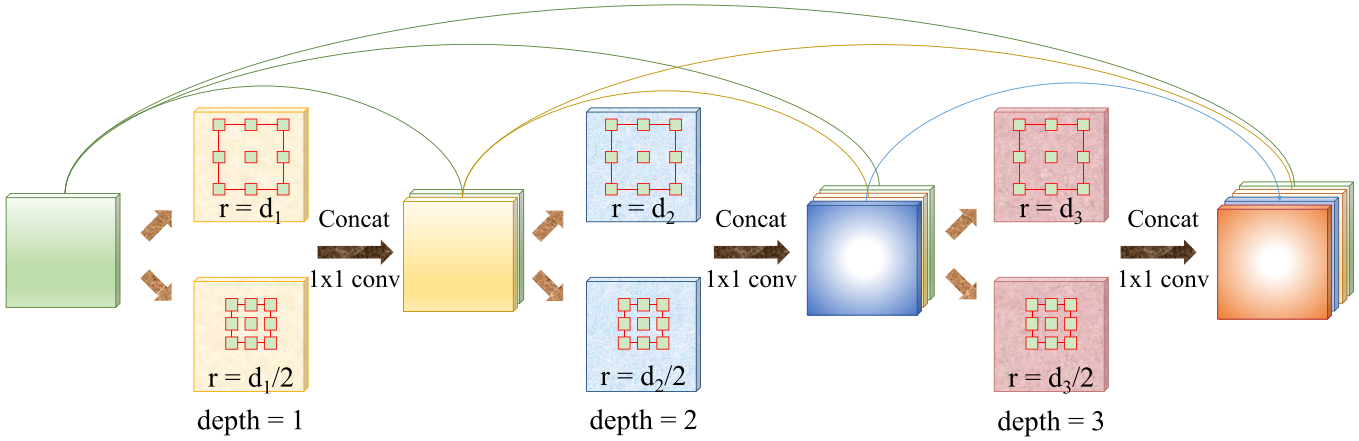
Fig. 3a shows the schematic illustration of our sub-region dilated block with a $k \times k$ region separations. Specifically, given a 3D feature map $\boldsymbol{M}$ (size: $h \times w \times c$) from a particular CNN layer, the sub-region block first separates $\boldsymbol{M}$ into $k \times k$ sub-regions, where the size of features at each sub-region is $(h/k) \times (w/k) \times c$. Then we design a parallel ASPP block (see Section III-B for details) at each sub-region to learn more obvious semantic features inside the sub-region, resulting in k^2 ($k^2 = 4$) feature maps (denoted as $\boldsymbol{F}_1$, $\boldsymbol{F}_2$, $\boldsymbol{F}_3, \ldots,$ and $\boldsymbol{F}_{k^2}$, respectively). Note that ASPP blocks do not share weights in any sub-region in order to make them independent After that, we employ two successive 3×3 convolutional layers on $\boldsymbol{M}$, and then a 1×1 convolutional layer to generate an attention map $\boldsymbol{W}$, which has k^2 channels. Each channel of $\boldsymbol{W}$ are denoted as $\boldsymbol{W}_1$, $\boldsymbol{W}_2, \ldots,$ and $\boldsymbol{W}_{k^2}$. Note that we adopt the ReLU activation function in the two 3×3 convolutional layers, and the Sigmoid activation function in the 1×1 convolutional layer.

Then, we multiply $\boldsymbol{W}_1$ and $\boldsymbol{F}_1$, $\boldsymbol{W}_2$ and $\boldsymbol{F}_2, \ldots,$ $\boldsymbol{W}_{k^2}$ and $\boldsymbol{F}_{k^2}$ in an element-wise manner, and compose these multiplication resultant features to form a new feature map ($\boldsymbol{J}$), which is then concatenated with the input feature map $\boldsymbol{M}$. Finally, we use a 3×3 convolutional layer on the resultant feature map to generate the output (denoted as $\widehat{\boldsymbol{M}}$) of our SRDB; see Fig. 3(a). Mathematically, $\widehat{\boldsymbol{M}}$ is computed as

$$\widehat{\boldsymbol{M}} = \Phi(\omega * Cat(\boldsymbol{M}, \boldsymbol{J}) + b) \tag{1}$$



(a) The schematic illustration of an example of our SRDB. Here, we divide the input feature map into $K \times K$ sub-regions and $K = 2$. As shown in Figure 2, K is set as 1 and 3 in other CNN layer. Please refer to Figure 3b for the details of the parallel ASPP. Note that $\boldsymbol{W}$ has four channels, and the four channels of $\boldsymbol{W}$ are corresponding to $\boldsymbol{W}_1$, $\boldsymbol{W}_2$, $\boldsymbol{W}_3$, and $\boldsymbol{W}_4$.



(b) The schematic illustration of an example our parallel ASPP block, which connected a series of atrous convolution layers with different dilated rates (denoted as r), and r = d_1, $\frac{d_1}{2}$, d_2, $\frac{d_2}{2}$, d_3, and $\frac{d_3}{2}$.

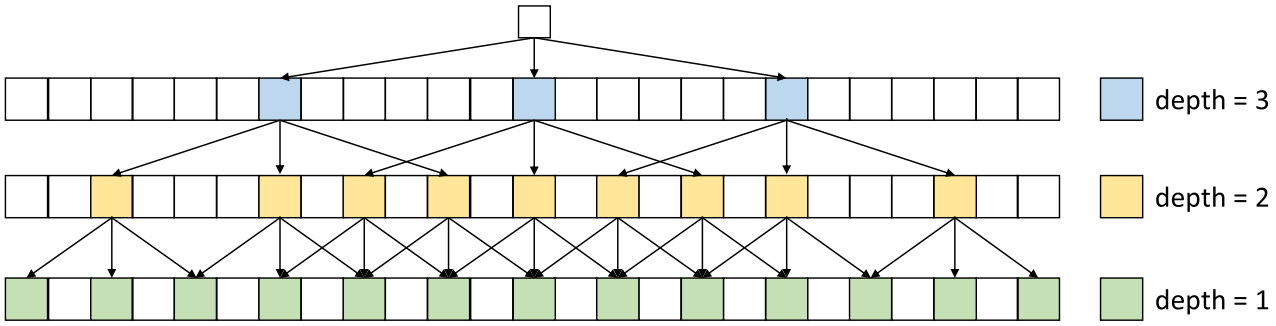
Fig. 3. The schematic illustration of the proposed sub-region dilated block (SRDB), which designs parallel ASPP blocks to learn multi-scale features from each sub-region. Figure 3a shows the basic structure and Fig. 3b illustrates the details of parallel ASPP module. "depth" means the dilated convolution layer.

where Cat is the concatenation operation across the channel direction; $*$ represents the convolution operation; ω and b are the weights and bias of the 3×3 convolutional layer; and Φ is the ReLU activation function in our implementation.

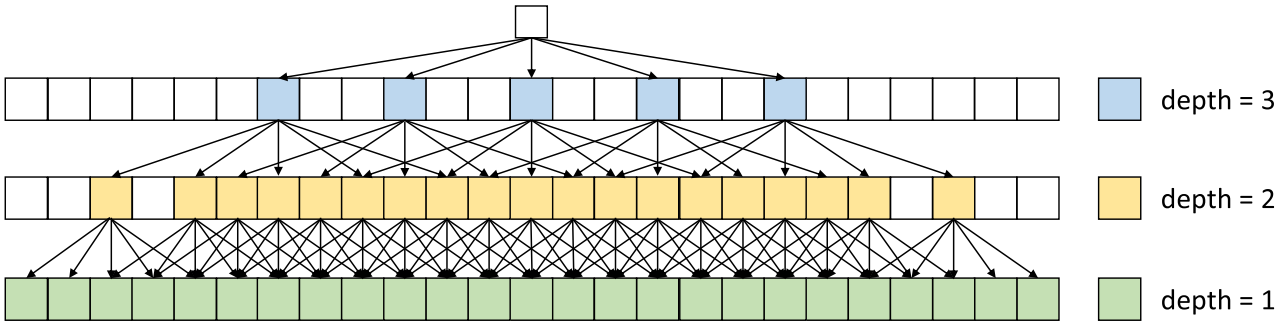
B. Parallel ASPP Block

Note that salient objects exhibit a very large scale change, which indicates that multi-scale features are required to cover the large scale range for inferring various salient objects. Chen *et al.* [54] proposed an atrous spatial pyramid pooling (ASPP) module for generating multi-scale features by concatenating multiple atrous convolution layers with different large dilation rates, since the dilated convolution can generate features with large receptive field but without sacrificing spatial resolution. More recently, Yang *et al.* [55] develop a DenseASPP to densely cover scale range by connecting atrous convolution layers in a dense manner [56]. Although the two methods above can capture multi-scale features that cover a large receptive field, both of them suffer from a"gridding issue" [57], which means that many positions in large receptive field windows are not used in the dilated convolutions, losing many spatial neighborhood information. Fig. 4a show an example of the gridding issue in the DenseASPP, and we can observe that many elements in each dilated convolutional layer are neglected for extracting multi-scale context features.

Note that most of feature positions are involved in the dilated convolution if the dilated rate is smaller. Inspired by this, we develop a parallel ASPP block to alleviate this gridding problem by imposing additional dilated convolutional layers (with small dilated rates) into each dilated layer (depth) of the ASPP. Fig. 3b shows the diagram of our parallel ASPP block. Specifically, we first employ three dilated convolutional layers, and dilated rates (denoted as r) are d_1, d_2, and d_3. Then, we add another dilated layer with a half dilated rate into each depth (dilated convolutional layer), and then densely connect them to form the output features of our parallel ASPP block. As shown in Figure 2, we apply our sub-region dilated block (SRDB) to refine features at the last three convolutional layers by balancing the time performance and saliency detection performance. And in each SRDB, we use the parallel ASPP



(a) The diagram of DenseASPP for the group of dilation rate: 2, 4, 6.



(b) The diagram of ParallelASPP for the group of dilation rate: 2, 4, 6.

Fig. 4. Comparison between DenseASPP and the parallel ASPP. Note that for each element at different dilated layers (depth), we use the while color for unused elements and other three colors for used elements in learning the multi-scale feature representation. Apparently, our parallel ASPP can cover more elements than the DenseASPP in each depth (dilated convolutional layer).

block with the same three dilated convolutional rates for each sub-region of the SRDB. We empirically set the three dilated rates (d_1, d_2, and d_3) in the last three CNN layers are set as (2 4 10), (2 4 8), and (2 4 6).

Our parallel ASPP block not only covers dense scale range, but also uses more neighboring feature positions (or pixels) for learning multi-scale feature representations for salient object detection. Fig. 4b visualizes an example of our parallel ASPP, which demonstrates that more elements (or feature positions) are utilized for learning multi-scale features when compared to the DenseASPP. Hence, our parallel ASPP can learn more powerful multi-scale feature representations than the DenseASPP, making the saliency map predictions more accurate (see Section IV-C for detailed comparisons).

C. Training and Testing Strategies

1) Training Parameters: In order to accelerate the training process and reduce the over-fitting issue, we use the well-trained DenseNet network on ImageNet [56] to initialize parameters of feature extraction network (see Fig. 2), while other layers are randomly initialized from a Gaussian distribution. We train our network on 2 GPUs (GTX 1080Ti) with a mini-batch size of 8 and stop the training process after 40k iterations. The stochastic gradient descent (SGD) algorithm is employed to optimize the loss function of the whole network by setting the momentum and the weight decay as 0.9 and 0.00001, respectively. The initial learning rate is set as 0.001 and it reduces by a factor of 0.1 at 15k iterations. We use an input size of 384 $\times$ 384 for each image in the training/testing stages, and images in the training dataset are randomly rotated, resized and horizontally flipped for data argumentation. Figure 5 shows the details of the feature extraction network of our method (see Figure 2).

2) Network Testing: In the testing stage, our network predicts only a saliency map from the last CNN layer (see Fig. 2) and use this prediction as the final result of our network. We utilize 8 hours to train our network and the testing time for a 400×400 image is about 0.067 second on a single GPU.

IV. Experimental Results

In this section, we will introduce the benchmark datasets and evaluation metrics, and present experiments to verify our DSR-Net. Our code, the trained models, and the predicted saliency maps on all five benchmark datasets are at https://github.com/Ball-Chen/DSR-Net.

A. Datasets and Evaluation Metrics

1) Benchmark Datasets: We used five widely-used saliency benchmark datasets in our experiments: (i) ECSSD [37] has 1,000 natural images, which have semantically meaningful but complex structures; (ii) PASCAL-S [51] consists of 850 images with several salient objects. (iii) HKU-IS [20] has 4,447 images with multiple salient objects; (iv) DUT-OMRON [38] has 5,168 images, and each image has one or more salient objects (v) DUTS [53] contains a training set of 10,553 images and a testing set (denoted as DUTS-test) of 5,019 images. Images in this dataset have

		operation	output size
Input image			384×384×3
Stage 1	conv	kernel size: 7×7, stride = 2, channel = 64, ReLU	192×192×64
Stage 2	pooling	kernel size: 3×3, maxpooling, stride = 2, ReLU	96×96×64
	dense block	$\begin{pmatrix} kernel\ size\colon 1\times 1, stride=1, channel=128, ReLU \\ kernel\ size\colon 3\times 3, stride=1, channel=128, ReLU \end{pmatrix}\times 6$	96×96×256
	transition block	kernel size: 1×1, stride=1, channel=128, ReLU	96×96×128
Stage 3	pooling	kernel size: 2×2 avg pooling, stride = 2, ReLU	48×48×128
	dense block	$\begin{pmatrix} kernel\ size\colon 1\times 1, stride=1, channel=128, ReLU \\ kernel\ size\colon 3\times 3\ conv, stride=1, channel=32, ReLU \end{pmatrix}\times 12$	48×48×512
	transition block	kernel size: 1×1, stride=1, channel=256, ReLU	48×48×256
Stage 4	pooling	kernel size: 2×2, avg pooling, stride = 2, ReLU	24×24×256
	dense block	$\begin{pmatrix} kernel\ size\colon 1\times 1, stride=1, channel=128, ReLU \\ kernel\ size\colon 3\times 3\ stride=1, channel=128, ReLU \end{pmatrix}\times 24$	24×24×1024
	transition block	kernel size: 1×1, stride=1, channel=512, ReLU	24×24×512

Fig. 5. The architecture details of the feature extraction network of our method. We use the pooling layer at each stage to reduce the size of feature maps.

TABLE I

COMPARISON WITH 26 STATE-OF-THE-ART METHODS IN TERMS OF F_β, S_m, AND MAE METRICS. THE TOP THREE RESULTS ARE HIGHLIGHTED IN **Red**, **Green**, AND **Blue**, RESPECTIVELY. NOTE THAT "−" DENOTES THE METRIC RESULTS ON THE CORRESPONDING BENCHMARK DATASET IS NOT PUBLICLY AVAILABLE, AND "*" MEANS THAT THE CONDITIONAL RANDOM FIELD (CRF) IS APPLIED AS THE POST-PROCESSING STEP IN THE SALIENCY DETECTION METHODS

Method	CRF	ECSSD 1,000 images			PASCAL-S 850 images			HKU-IS 4,447 images			DUTS-TE 5,019 images			DUT-OMRON 5,168 images		
		F_β	S_m	MAE	F_β	S_m	MAE	F_β	S_m	MAE	F_β	S_m	MAE	F_β	S_m	MAE
DSR-Net (ours)	No	**0.952**	**0.924**	0.039	**0.898**	**0.806**	**0.071**	**0.939**	**0.915**	0.035	**0.895**	**0.871**	0.043	**0.833**	**0.832**	0.060
DSR-Net* (ours)	Yes	**0.950**	**0.922**	**0.031**	**0.888**	**0.798**	**0.068**	**0.939**	**0.915**	**0.027**	**0.891**	**0.863**	**0.036**	0.811	**0.829**	**0.053**
PoolNet-R [27]	No	**0.944**	**0.921**	0.039	0.865	**0.794**	0.080	**0.934**	**0.912**	**0.033**	**0.886**	**0.871**	**0.040**	**0.830**	**0.836**	**0.056**
BASNet [28]	No	0.942	0.916	**0.037**	0.858	0.785	0.084	**0.929**	**0.909**	**0.032**	0.860	0.853	0.047	0.811	**0.836**	**0.056**
CPD-R [58]	No	0.939	0.918	**0.037**	0.861	0.789	0.078	0.925	0.906	0.034	0.865	**0.858**	0.043	0.797	0.825	**0.056**
AFNet [30]	No	0.935	0.917	0.042	0.866	0.792	**0.076**	0.925	0.905	0.036	0.867	0.855	0.045	**0.820**	0.826	0.057
MLMSNet [29]	No	0.930	0.909	0.045	0.858	0.790	0.079	0.922	0.906	0.039	0.854	0.851	0.048	0.793	0.809	0.064
Pi-RC [25]	Yes	0.940	0.916	**0.035**	**0.883**	0.790	0.077	0.927	0.905	0.031	0.866	0.849	**0.041**	0.804	0.826	**0.054**
RAS [47]	Yes	0.916	0.893	0.058	0.842	0.772	0.122	0.913	0.887	0.045	0.831	0.838	0.059	0.785	0.814	0.063
C2S [59]	Yes	0.911	0.896	0.053	0.845	0.793	0.084	0.898	0.889	0.046	0.811	0.831	0.062	0.759	0.799	0.072
R^3Net [24]	Yes	0.935	0.908	0.040	0.845	0.750	0.100	0.916	0.895	0.036	0.833	0.817	0.058	0.805	0.812	0.063
PAGRN [26]	Yes	0.928	0.889	0.044	0.862	0.792	0.094	0.918	0.887	0.048	0.854	0.837	0.055	0.771	0.775	0.071
DGRL [49]	Yes	0.925	0.906	0.045	0.850	0.796	0.080	0.914	0.897	0.037	0.834	0.845	0.051	0.785	0.810	0.060
LPS [60]	Yes	0.910	0.888	0.054	0.805	0.786	0.093	0.903	0.874	0.033	0.800	0.797	0.054	0.734	0.786	0.070
RADF [23]	Yes	0.924	0.894	0.049	0.832	0.754	0.102	0.914	0.889	0.039	0.819	0.814	0.061	0.789	0.815	0.060
SRM [61]	Yes	0.917	0.895	0.054	0.847	0.782	0.085	0.906	0.887	0.046	0.827	0.835	0.059	0.769	0.798	0.069
DSS [22]	Yes	0.916	0.882	0.053	0.836	0.777	0.096	0.910	0.881	0.041	0.825	0.822	0.057	0.771	0.788	0.066
Amulet [44]	Yes	0.915	0.894	0.059	0.837	0.794	0.098	0.895	0.883	0.052	0.778	0.803	0.085	0.743	0.781	0.090
UCF [43]	Yes	0.911	0.883	0.078	0.828	0.792	0.126	0.886	0.875	0.074	0.771	0.777	0.117	0.771	0.758	0.117
NLDF [41]	Yes	0.905	0.875	0.063	0.831	0.790	0.099	0.902	0.879	0.048	0.812	0.815	0.066	0.753	0.770	0.080
DHSNet [62]	Yes	0.907	0.884	0.059	0.829	0.788	0.094	0.890	0.881	0.053	0.807	0.836	0.067	-	-	-
DCL [45]	Yes	0.890	0.868	0.088	0.805	0.783	0.125	0.885	0.861	0.072	0.782	0.795	0.088	0.739	0.764	0.157
ELD [63]	Yes	0.867	0.841	0.079	0.773	-	0.123	0.839	-	0.074	0.738	0.719	0.093	0.719	0.751	0.091
RFCN [42]	Yes	0.890	0.860	0.107	0.837	0.793	0.118	0.892	0.859	0.079	0.784	0.791	0.091	0.738	0.774	0.095
LEGS [64]	Yes	0.827	0.787	0.118	0.762	0.682	0.155	0.766	-	0.119	0.655	-	0.138	0.669	-	0.133
MDF [20]	Yes	0.832	0.776	0.105	0.768	0.672	0.146	-	-	-	0.730	0.727	0.094	0.694	0.721	0.092
DRFI [18]	No	0.786	-	0.164	0.698	-	0.207	0.777	-	0.145	0.647	-	0.175	-	-	-
BSCA [65]	No	0.758	0.725	0.182	0.667	0.633	0.223	0.719	0.700	0.175	0.597	0.630	0.197	0.616	0.652	0.191

multiple salient objects with different region sizes. Following recent works [25], [26], [48], [49], we use the training set of DUTS [53] for training our network.

2) Evaluation Metrics: We adopt several widely-used metrics to quantitatively evaluate the performance of different saliency models, and they are the precision-recall curves (denoted as PR curves), F-measure (denoted as F_β), S-measure (denoted as S_m), mean absolute error (denoted as MAE), weighted F-measure (denoted as wF_β [66]), AUC [67], and E-measure (denoted as E_m [68]). Overall, a better saliency detector shall have a larger F_β, a larger S_m, a smaller MAE, a larger wF_β, and a larger S_m.

TABLE II

COMPARISON WITH MORE RECENT STATE-OF-THE-ART METHODS IN TERMS OF wF_β AND E_m METRICS. THE TOP THREE RESULTS ARE HIGHLIGHTED IN **Red**, **Green**, AND **Blue**, RESPECTIVELY. NOTE THAT "–" DENOTES THE METRIC RESULTS ON THE CORRESPONDING BENCHMARK DATASET IS NOT PUBLICLY AVAILABLE, AND "*" MEANS THAT THE CRF IS APPLIED AS THE POST-PROCESSING STEP IN THE SALIENCY DETECTION METHODS

Method	CRF	ECSSD 1,000 images			PASCAL-S 850 images			HKU-IS 4,447 images			DUTS-TE 5,019 images			DUT-OMRON 5,168 images		
		wF_β	AUC	E_m	wF_β	AUC	E_m	wF_β	AUC	E_m	wF_β	AUC	E_m	wF_β	AUC	E_m
DSR-Net (ours)	No	0.891	0.9514	0.805	0.780	0.9003	0.787	0.872	0.9529	0.798	0.786	0.9303	0.725	0.708	0.8978	0.677
DSR-Net* (ours)	Yes	**0.918**	**0.9687**	**0.953**	**0.796**	**0.9249**	**0.850**	**0.905**	**0.9743**	**0.954**	**0.825**	**0.9651**	**0.918**	**0.747**	**0.9339**	**0.861**
PoolNet-R [27]	No	0.896	0.9626	0.848	0.773	0.9108	0.811	0.878	0.9665	0.859	0.797	0.9586	0.781	0.725	0.9260	0.739
BASNet [28]	No	0.904	0.9471	0.938	0.774	0.8885	0.834	0.880	0.9538	0.935	0.793	0.9311	0.886	0.751	0.9101	0.857
CPD-R [58]	No	0.898	0.9583	0.902	0.767	0.9020	0.827	0.875	0.9607	0.888	0.787	0.9428	0.837	0.719	0.9090	0.788
AFNet [30]	No	0.886	0.9580	0.849	0.777	0.9083	0.810	0.869	0.9651	0.839	0.776	0.9480	0.785	0.714	0.9173	0.760
MLMSNet [29]	No	0.871	0.9602	0.830	0.766	0.9086	0.796	0.859	0.9691	0.826	0.754	0.9544	0.761	0.680	0.9074	0.761
Pi-RC [25]	Yes	0.908	0.9502	0.948	0.784	0.8950	0.847	0.890	0.9505	0.946	0.800	0.9235	0.907	0.743	0.8977	0.860
C2S [59]	Yes	0.854	0.9560	0.861	0.763	0.9136	0.819	0.835	0.9652	0.844	0.713	0.9434	0.767	0.663	0.9164	0.730
R^3Net [24]	Yes	0.901	0.9454	0.942	0.735	0.8673	0.802	0.877	0.9526	0.934	0.751	0.9031	0.872	0.721	0.9003	0.837
PAGRN [26]	Yes	0.834	0.9602	0.559	0.693	0.8847	0.592	0.819	0.9558	0.506	0.715	0.9295	0.613	0.622	0.8723	0.604
DGRL [49]	Yes	0.883	0.9510	0.868	0.780	0.9003	0.809	0.865	0.9564	0.856	0.753	0.9294	0.789	0.697	0.8990	0.754
LPS [60]	Yes	0.853	0.9507	0.838	0.764	0.9086	0.783	0.837	0.9483	0.861	0.703	0.9262	0.764	0.673	0.9171	0.720
RADF [23]	Yes	0.883	0.9363	0.922	0.741	0.8660	0.802	0.872	0.9454	0.920	0.740	0.9165	0.846	0.722	0.9142	0.828
SRM [61]	Yes	0.853	0.9572	0.817	0.745	0.9102	0.776	0.835	0.9648	0.800	0.714	0.9427	0.710	0.658	0.9130	0.677
DSS [22]	Yes	0.870	0.9232	0.921	0.726	0.8464	0.796	0.865	0.9317	0.929	0.746	0.8993	0.871	0.695	0.8542	0.838
Amulet [44]	Yes	0.848	0.9317	0.917	0.755	0.9224	0.680	0.766	0.9709	0.666	0.655	0.9461	0.576	0.654	0.9042	0.781
UCF [43]	Yes	0.789	-	0.456	0.731	-	0.617	0.779	-	0.577	0.596	-	0.379	0.564	-	0.345
NLDF [41]	Yes	0.839	0.9390	0.881	0.727	0.9224	0.837	0.838	0.9533	0.885	0.701	0.9271	0.777	0.634	0.8952	0.735

Given a predicted saliency map (denoted as $\mathcal{D}$), we can obtain binarized saliency maps with a set of thresholds in [0, 1] and produce a pair of precision and recall scores by comparing each binarized map against the ground truth (denoted as $\mathcal{G}$). The precision scores compute the percentage of salient pixels being correctly detected while the recall scores show the ratio between detected salient pixels and salient pixels in the ground truth. The PR curve describes the model performance by plotting all the pairs of averaged precision and recall pairs of all images in the dataset; see Fig. 7 for comparisons among different saliency detections on the five benchmark datasets. The AUC (Area Under ROC Curve) score can be computed from a receiver operating characteristic (ROC) curve, which is estimated according to the true positive rates and false positive rates obtained during plotting the PR curve; see [67] for AUC details. F-measure (F_β) balances the average precision and average recall over saliency maps of all images in the dataset:

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

where $\beta^2 = 0.3$; see [22], [69]. Instead of plotting the whole F-measure, we follow existing works [22], [24], [25] to directly use the maximal F_β for comparisons.

S-measure [70] (S_m) quantitatively compare $\mathcal{D}$ against $\mathcal{G}$ by considering its object-aware and region-aware structural similarities:

$$S_m = \alpha \times S_o(\mathcal{D}, \mathcal{G}) + (1-\alpha) \times S_r(\mathcal{D}, \mathcal{G}) \tag{3}$$

where $S_o(\mathcal{D}, \mathcal{G})$ and $S_r(\mathcal{D}, \mathcal{G})$ denote the object-aware and region-aware structural similarities, respectively; see [70]. $\alpha = 0.5$, as suggested in [70].

TABLE III

COMPARISON WITH THE STATE-OF-THE-ARTS ON THE RECENT SOC DATASET [72]. THE TOP THREE RESULTS ARE HIGHLIGHTED IN **Red**, **Green**, AND **Blue**. "*" MEANS THAT THE CRF IS USED AS THE POST-PROCESSING

Method	SOC			
	F_β	S_m	wF_β	MAE
Ours (DSR-Net)	0.8145	0.7967	0.6828	0.0913
Ours (DSR-Net*)	0.7979	0.7914	0.7131	0.0838
Deepside-joint-fusion [71]	0.7391	0.7813	0.6648	0.0933
R^3Net [24]	0.6960	0.6820	0.7480	0.1350
DGRL [49]	0.7200	0.7728	0.6211	0.1012
SRM [61]	0.7071	0.7632	0.6147	0.1074
NLDF [41]	0.6663	0.7234	0.5774	0.1285
RFCN [42]	0.6581	0.7180	0.5797	0.1276
DHS [62]	0.6844	0.7354	0.6103	0.1123
DSS [22]	0.6284	0.6726	0.5625	0.1411
DCL [45]	0.6440	0.6960	0.5570	0.1373

MAE [34] computes the similarity of $\mathcal{D}$ and $\mathcal{G}$ by directly averaging the pixel-wise absolute difference between them:

$$MAE = \frac{1}{\mathbf{U} \times \mathbf{V}} \sum_{u=1}^{\mathbf{U}} \sum_{v=1}^{\mathbf{V}} \|\mathcal{D}(u,v) - \mathcal{G}(u,v)\|, \tag{4}$$

where $\mathbf{U}$ and $\mathbf{V}$ are the width and height of $\mathcal{G}$.

Weighted F-measure [66] (wF_β) provides a generalized F-measure by extending the precision and recall scores to real values with different weights:

$$wF_\beta = \frac{(1+\gamma^2) \times Precision^w \times Recall^w}{\gamma^2 \times Precision^w + Recall^w}, \tag{5}$$

where $Precision^w$ and $Recall^w$ are the weighted precision and recall scores and weights assigned at different

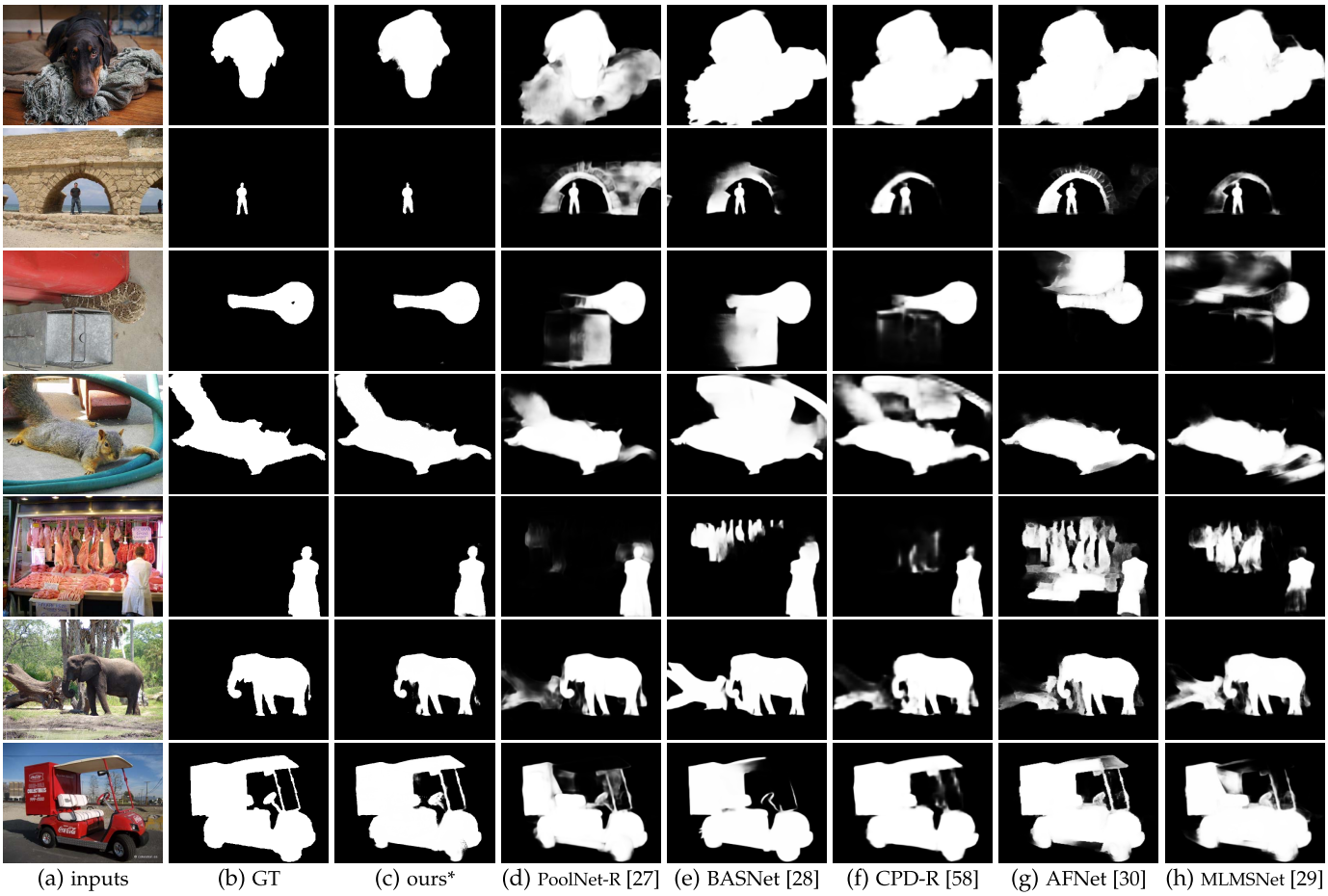


Fig. 6. Visual comparison of saliency map results produced by different methods. (a) Input images (one salient objects) with different complex scenarios; (b) Ground truths (denoted as 'GT'); (c)-(h) Saliency maps predicted by our method, PoolNet-R [27], BASNet [28], CPD-R [58], AFNet [30], and MLMSNet [29]. Apparently, our network produces more accurate saliency maps than other methods. Note that "*" denotes the CRF is used as the post-processing in the methods.

locations are computed by considering neighborhood information. $\gamma^2 = 0.3$, as suggested in [66], [71].

E-measure (E_m) [68] compares $\mathcal{D}$ and $\mathcal{G}$ by simultaneously considering the global means of the image and the local pixel matching:

$$E_m = \frac{1}{\mathbf{U} \times \mathbf{V}} \sum_{u=1}^{\mathbf{U}} \sum_{v=1}^{\mathbf{V}} \Theta(u, v), \tag{6}$$

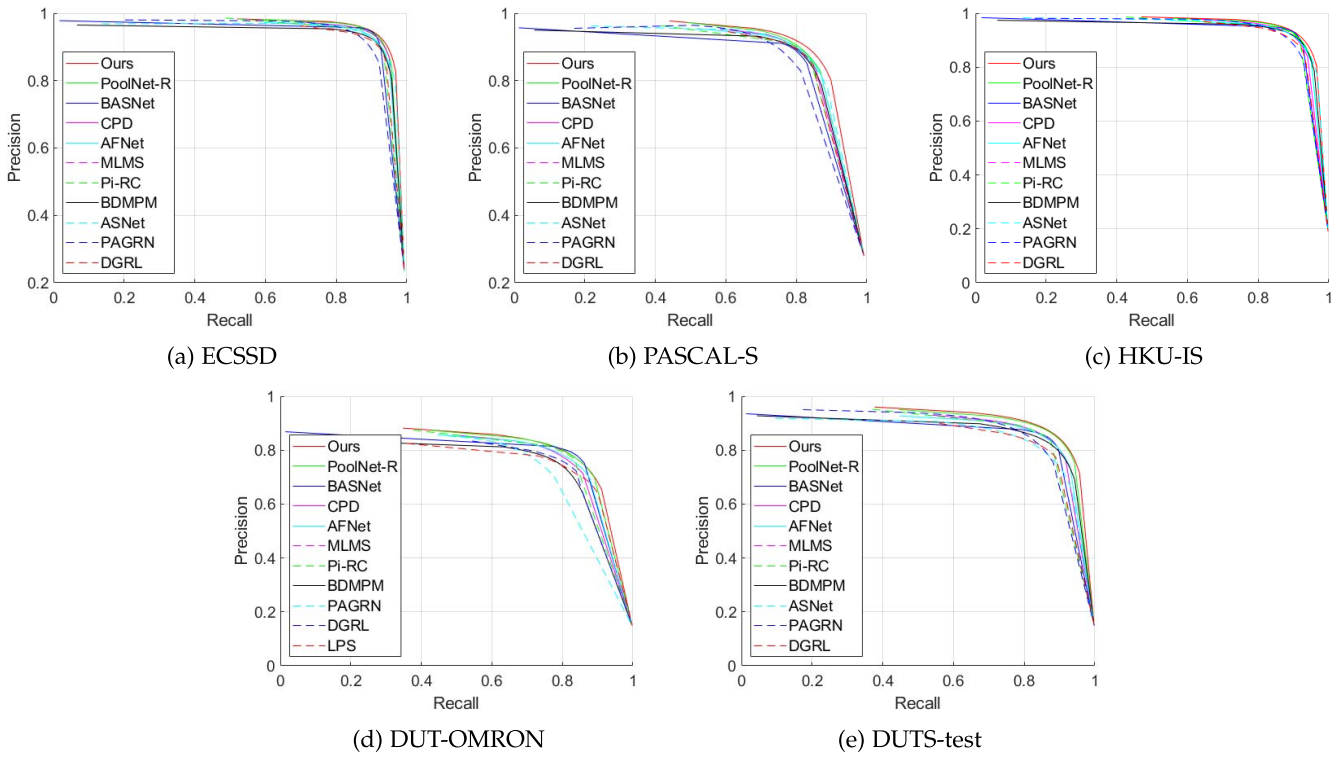
where Θ denotes the enhanced alignment matrix, which represents the correlation between $\mathcal{D}$ and $\mathcal{G}$; please see [68] for the details of computing Θ.

B. Comparison With the State-of-the-Arts

We evaluate the effectiveness of our network by comparing it against 26 state-of-the-art salient object detectors; see the first column in Table I for the list of our competitors. Among the 26 methods, BSCA [65] and DRFI [18] utilize hand-crafted features to infer the salient objects, while other methods formulate different network models to learn the deep convolutional features for identifying salient objects from the single input image. To make the comparisons fair, we obtained the saliency maps of all 26 competitors either from the authors or by using their implementations with the released training models and parameters.

1) Quantitative Comparison: Table I summaries the F_β, S_m and MAE results of our method and 26 competitors in all the five benchmark datasets, while Table II compares $w F_\beta$, AUC, and E_m results of different methods. Apparently, our DSR-Net consistently outperforms other saliency detectors on almost all the five benchmark datasets. Furthermore, our method with CRF has more accurate results than all compared saliency detectors on the three largest datasets ("HKU-IS", "DUT-OMRON" and "DUTS-test"), which contain more complex and multiple salient objects. It demonstrates that our method can better handle challenging cases of saliency detection; please refer to Figure 6 and Figure 8 for visual comparisons of different predicted saliency maps.

2) Visual Comparison: We also visualize the saliency maps produced by our network and state-of-the-art methods; see Figures 6 and 8 for examples. As shown in these two figures, other methods (d)-(h) tend to include non-salient backgrounds or lose salient details in their predicted saliency maps, while our DSR-Net predicts more accurate saliency maps (c), which are more consistent with the ground truths (b). Furthermore, for those challenging cases with small salient



(a) ECSSD (b) PASCAL-S (c) HKU-IS

(d) DUT-OMRON (e) DUTS-test

Fig. 7. Visual comparison of the precision-recall (PR) curves on the five widely-used saliency detection datasets: (a) ECSSD; (b) PASCAL-S; (c) HKU-IS; (d) DUT-OMRON; and (e) DUTS-test.

TABLE IV

COMPONENT ANALYSIS IN TERMS OF F_β, S_m, AND MAE METRICS. NOTE THAT "SR" DENOTES "SUB-REGION," AND "PB" DENOTES THE USE OF PARALLEL ASPP BLOCK. "*" MEANS THAT THE CRF IS USED AS THE POST-PROCESSING

	Sub-regions	Parallel-Block	Training time (hours)	ECSSD			PASCAL-S			HKU-IS			DUTS-test			DUT-OMRON		
				F_β	S_m	MAE	F_β	S_m	MAE	F_β	S_m	MAE	F_β	S_m	MAE	F_β	S_m	MAE
Basic*	×	×	2.13	0.943	0.908	0.039	0.884	0.781	0.077	0.932	0.900	0.032	0.881	0.849	0.039	0.800	0.816	0.055
Ours-w/o-PB*	√	×	4.09	0.948	0.919	0.033	0.886	0.792	0.068	0.938	0.913	0.028	0.889	0.861	0.037	0.808	0.826	0.054
Ours-w/o-SR*	×	√	2.76	0.948	0.918	0.033	0.886	0.793	0.068	0.938	0.912	0.027	0.889	0.859	0.044	0.806	0.824	0.054
Ours-denseASPP*	√	×	4.27	0.934	0.899	0.043	0.870	0.781	0.076	0.922	0.894	0.033	0.855	0.827	0.044	0.763	0.787	0.063
DSR-Net* (ours)	√	√	3.91	**0.950**	**0.922**	**0.031**	**0.888**	**0.798**	**0.068**	**0.939**	**0.915**	**0.027**	**0.891**	**0.863**	**0.036**	**0.811**	**0.829**	**0.053**

objects (1st∼3rd rows in Figure 8), and multiple salient objects (4-th∼10-th rows in Figure 8), our network also produces better saliency detection results over our competitors. It indicates that learning sub-region features in our network can highly suppress the saliency inference corruptions from non-salient objects, which are coupled with salient ones in other methods for detecting salient objects. *Please refer to the supplementary material for more visual comparisons.*

3) PR Curves: Apart from the five quantitative metrics (F_β, S_m, MAE, wF_β, and E_m), Fig. 7 compares the PR curves of different saliency detectors on the five common saliency detection benchmark datasets to further evaluate the effectiveness of the developed network. By observing PR curves of different networks, our method (red ones) has a better PR curve performance than all competitors. Moreover, as the recall score goes to 1, our method has larger precision value than other competitors, which demonstrates that our method achieves lower false positives.

4) SOC Dataset: Recently, Fan *et al.* [72] released a more challenging SOC dataset about saliency detection. Following [71], we test the trained network on $2,400$ images of the SOC. Table III reports the four quantitative metrics of our network and other saliency detectors on the SOC dataset. Apparently, our method has a superior metric performance over compared saliency detectors, demonstrating that our method more accurately identifies saliency objects of the SOC dataset.

C. Ablation Study

We conducted an ablation study experiment to verify the effectiveness of two major components of our network by considering four baseline networks. The first baseline (denoted as "Basic") is to remove the sub-region module and the parallel ASPP module from our network. The second one (denoted as "Ours-w/o-PB") is constructed by removing the parallel ASPP module from the whole network while the third one (denoted as "Ours-w/o-SR") is built by eliminating the sub-region module from our method. The last baseline (denoted as "Ours-denseASPP") is to replace our parallel ASPP module with the original dense ASPP module to verify the parallel ASPP module.

We test the four baselines and our method on all the five benchmark datasets, and Table IV and Table V report the quantitative comparisons in terms of five metrics including

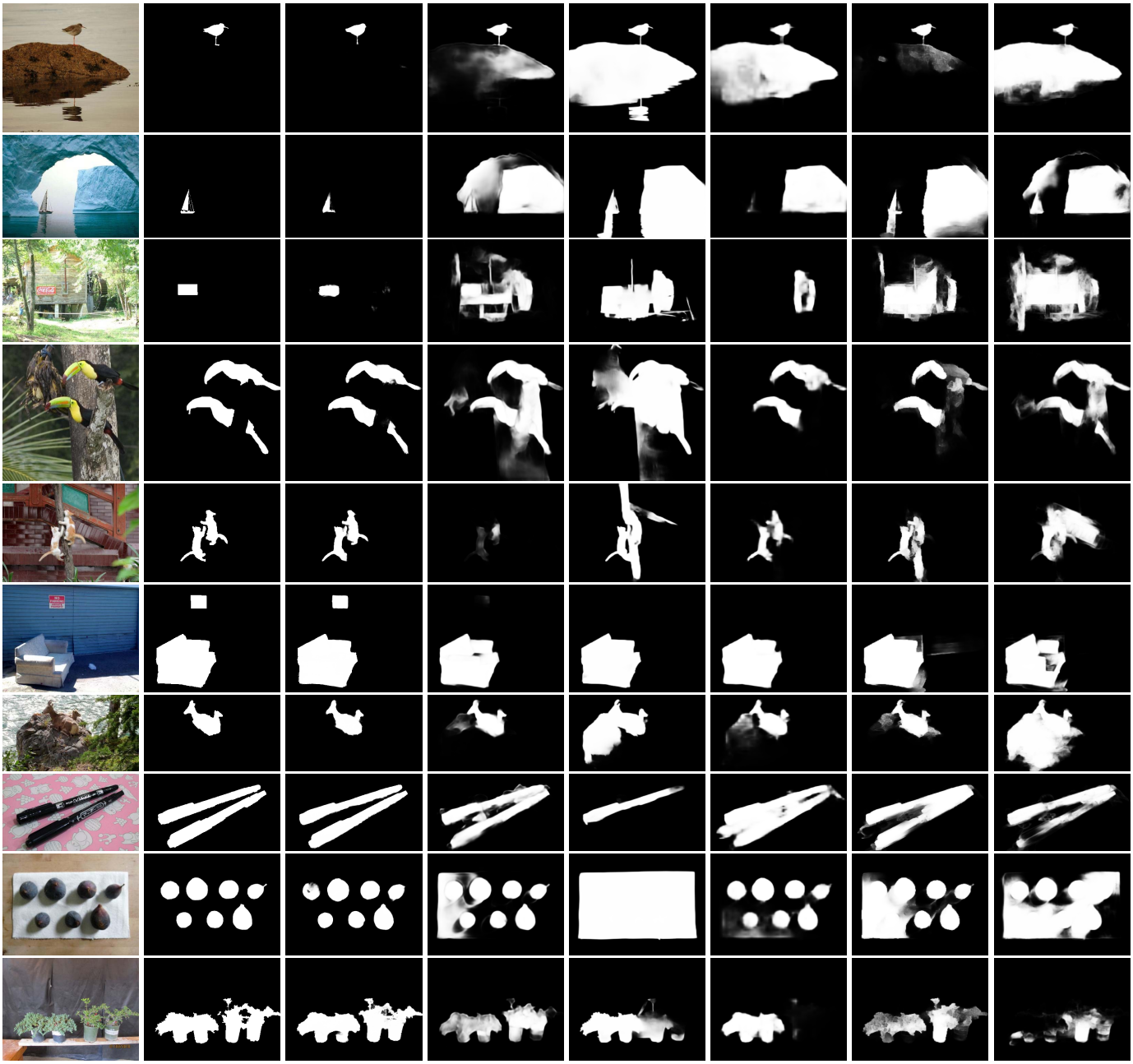


Fig. 8. More visual comparison results on input photos, which have small or multiple salient objects.

TABLE V

COMPONENT ANALYSIS IN TERMS OF wF_β, AUC, AND E_m METRICS. NOTE THAT "SR" DENOTES "SUB-REGION," AND "PB" DENOTES THE USE OF PARALLEL ASPP BLOCK. "*" MEANS THAT THE CRF IS USED AS THE POST-PROCESSING

	Sub-regions	Parallel-Block	Training time (hours)	ECSSD			PASCAL-S			HKU-IS			DUTS-test			DUT-OMRON		
				wF_β	AUC	E_m	wF_β	AUC	E_m	wF_β	AUC	E_m	wF_β	AUC	E_m	wF_β	AUC	E_m
Basic*	×	×	2.13	0.903	0.9456	0.947	0.773	0.8890	0.843	0.890	0.9462	0.951	0.805	0.9238	0.913	0.728	0.8876	0.860
Ours-w/o-PB*	√	×	4.09	0.915	0.9517	0.950	0.788	0.8968	0.845	0.904	0.9541	0.953	0.822	0.9310	0.917	0.742	0.8974	0.858
Ours-w/o-SR*	×	√	2.76	0.914	0.9508	0.950	0.790	0.8977	0.848	0.902	0.9537	0.954	0.819	0.9308	0.917	0.740	0.8967	0.861
Ours-denseASPP*	√	×	4.27	0.888	0.9364	0.935	0.772	0.8859	0.842	0.878	0.9388	0.941	0.767	0.9309	0.887	0.680	0.8653	0.825
DSR-Net* (ours)	√	√	3.91	**0.918**	**0.9518**	**0.953**	**0.796**	**0.9014**	**0.850**	**0.905**	**0.9543**	**0.954**	**0.825**	**0.9315**	**0.918**	**0.747**	**0.9005**	**0.861**

F_β, S_m, MAE, wF_β, AUC, and E_m. Obviously, our method has a better performance than "Ours-w/o-PB" and "Ours-w/o-SR", demonstrating that both our sub-region module and the parallel ASPP module contribute the superior saliency detection results of our network. Furthermore, our method predicts more accurate saliency maps than "Ours-denseASPP",

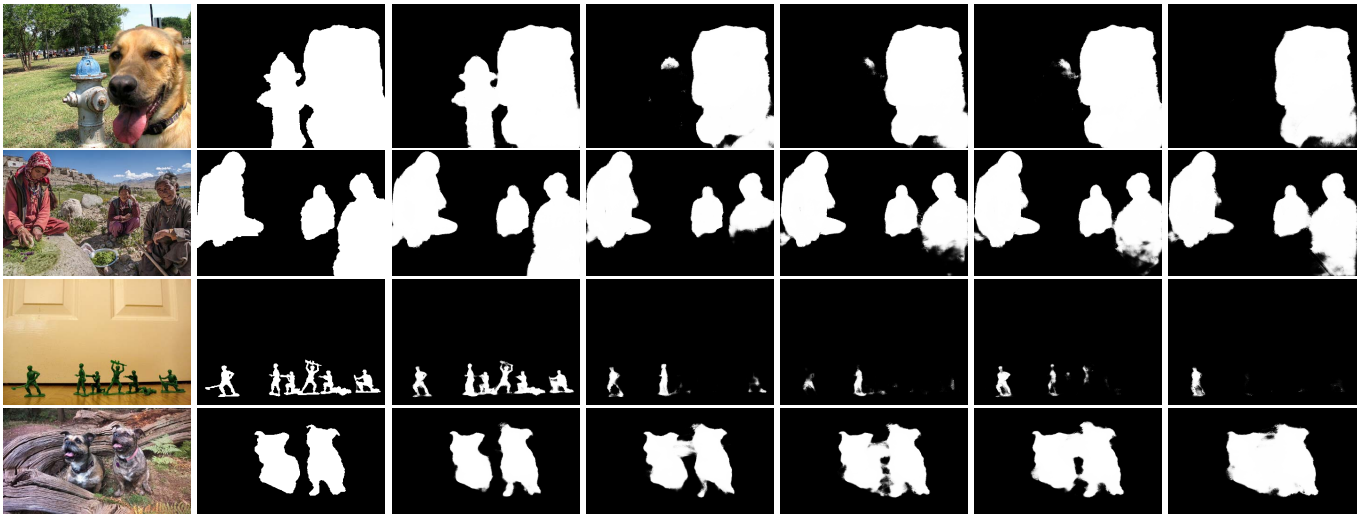


Fig. 9. Visual comparisons on saliency detection results produced by different networks in the ablation study experiment.

TABLE VI

ANALYSIS ON THE NUMBER OF THE SRDB IN TERMS OF F_β, AND MAE METRICS. "*" MEANS THAT THE CRF IS USED AS THE POST-PROCESSING

	SRDB number	Training time (hours)	ECSSD		PASCAL-S		HKU-IS		DUTS-test		DUT-OMRON	
			F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE
ours-2srdb*	2	2.96	0.9483	0.0319	0.8871	0.0687	0.9359	0.0278	0.8861	0.0369	0.8073	0.0542
ours-4srdb*	4	6.41	0.9475	0.032	0.8858	0.0688	0.9342	0.0281	0.8836	0.0370	0.8047	0.0539
DSR-Net* (ours)	3	3.91	**0.950**	**0.031**	**0.888**	**0.068**	**0.939**	**0.027**	**0.891**	**0.036**	**0.811**	**0.053**

TABLE VII

ANALYSIS ON THE NUMBER OF DILATED LAYERS OF THE PARALLEL ASPP MODULE IN TERMS OF F_β, AND MAE METRICS

	dilated layer number	Training time (hours)	ECSSD		PASCAL-S		HKU-IS		DUTS-test		DUT-OMRON	
			F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE
ours-2dl*	2	3.16	0.9474	0.0333	0.8843	0.0703	0.9380	0.0279	0.8820	0.0393	0.8084	0.0576
ours-4dl*	4	5.10	0.9488	0.328	0.8852	0.0695	0.9377	0.0280	0.8863	0.0374	0.8079	0.0548
DSR-Net* (ours)	3	3.91	**0.950**	**0.031**	**0.888**	**0.068**	**0.939**	**0.027**	**0.891**	**0.036**	**0.811**	**0.053**

TABLE VIII

COMPUTATIONAL COST ANALYSIS OF THE ABLATION STUDY

models	input size	parameter number	model memory (MB)	FLOPs (GFLOPs)
Basic*	384× 384	18, 872, 513	73	54.154
Ours-w/o-PB*	384× 384	40, 117, 185	155	83.541
Ours-w/o-SR*	384× 384	40, 622, 017	156	122.274
Ours-denseASPP*	384× 384	5, 641, 601	23	31.209
DSR-Net* (ours)	384× 384	75, 290, 703	290	165.367

TABLE IX

COMPARISON WITH THE STATE-OF-THE-ARTS ON THE MSD MIRROR DETECTION DATASET

Method	MSD	
	F_β	Acc
DSS [22]	0.743	0.665
Pi-RC [25]	0.808	0.844
RAS [47]	0.758	0.695
MirrorNet [73]	0.841	0.932
Ours (DSR-Net)	0.826	0.908

which shows that our parallel ASPP module learns a more powerful feature representations than the original Dense ASPP for salient object detection.

In addition, Figure 9 visualizes the saliency maps predicted by our method and the four baseline networks. As shown in this figure, the four baselines tend to include many non-salient objects or miss salient object boundaries when inferring salient objects. On contrast, our method can predict more accurate saliency maps, which are most consistent with the ground truth (see Figure 9(b)).

D. Parameter Discussion

1) Computational Cost: Table VIII presents an analysis on the computational cost of all the experimental setting of the ablation study. Apparently, the model size is increased when adding the subregions and the parallel ASPP modules into our network. However, as shown in Tables IV and V, the SRDB and the parallel ASPP modules enhance the saliency detection performance in the five benchmark datasets.

2) The Number of SRDB: We first construct two networks by setting the number of SRDB blocks in our network as 2 and 4, which are denoted as "ours-2srdb" and "ours-4srdb". Table VI reports the quantitative results on the five benchmark datasets. Apparently, the training time of our method is larger than "ours-4srdb", and our training time is smaller than "ours-2srdb". Meanwhile, our method outperforms "ours-2srdb" and

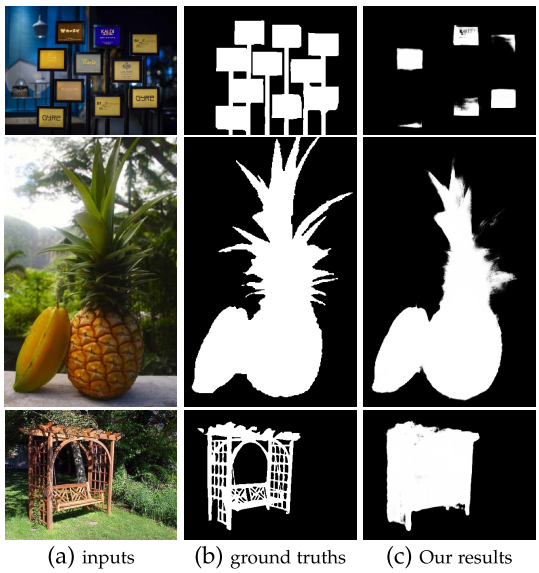


Fig. 10. Three failure cases of our method.

"ours-4srdb" in F_β and MAE, demonstrating that our method can better detect salient objects. Hence, we empirically set the number of SRDB as 3 in our DSR-Net.

3) The Number of Dilated Layers in Parallel ASPP: Another two networks are constructed by setting the number of dilated layers of the parallel ASPP as 2 (denoted as "ours-2dl") and 4 (denoted as "ours-4dl"). As shown in Table VII, our method also has a superior F_β and MAE performance than "ours-2dl" and "ours-4dl" on the five benchmark datasets, which demonstrates that more saliency detection accuracy is achieved when setting the dilated layer number as 3. As a result, we empirically use three dilated layers in all the parallel ASPP modules of our method.

E. More Discussions

1) Failure Cases: Although obtaining superior saliency detection performance on the five benchmarks, our method also has the failure cases, for which we found to be challenging also for the state-of-the-art salient object detectors. For instance, our method may fail for (i) multiple salient objects in very different scales (see Figure 10 (top)), where the network may lose several objects; (ii) salient objects with complex salient object boundaries (see Figure 10 (middle)), where there are insufficient context to fully detect those boundaries; and (iii) salient objects with many non-salient inner holes (see Figure 10 (bottom)), where our method may regard those inner holes as salient objects. Addressing those failure cases is regarded as a future direction of our work.

2) Application: Our network also has its applications. Here, we take the mirror detection as an example. Note that MirrorNet [73] has released a MSD dataset of the mirror detection. Following MirrorNet, we retrain our network on the MSD training set and test the trained model on the MSD testing set for comparisons. Table IX shows the metric values (in terms of F_β and Acc) of our network and other methods (including three saliency detectors and the MirrorNet). As can be seen, our method has a superior metric performance over other saliency detectors for the mirror detection application, but has not better metric results than MirrorNet, which is dedicated for the mirror detection by considering the mirror characteristics.

V. Conclusion

This paper presents a novel deep neural network for boosting the saliency detection from an input image. Our key idea is to separate the whole image into several sub-regions to eliminate the interference from many non-salient regions of the input image, design a parallel ASPP module to enlarge the receptive field for predicting the saliency maps in each sub-region, and then aggregate the saliency predictions from all the sub-regions for generating final saliency predictions of our network. Experiments on five widely-used saliency detection benchmark datasets demonstrate that our method consistently outperforms recent state-of-the-art saliency detectors on all the benchmark datasets.

References

[1] J. Han, S. He, X. Qian, D. Wang, L. Guo, and T. Liu, "An object-oriented visual saliency detection framework based on sparse coding representations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 12, pp. 2009–2021, Dec. 2013.

[2] J.-S. Kim, J.-Y. Sim, and C.-S. Kim, "Multiscale saliency detection using random walk with restart," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 2, pp. 198–210, Feb. 2014.

[3] J. Han, D. Zhang, X. Hu, L. Guo, J. Ren, and F. Wu, "Background prior-based salient object detection via deep reconstruction residual," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 8, pp. 1309–1321, Aug. 2015.

[4] Y. Yuan, C. Li, J. Kim, W. Cai, and D. D. Feng, "Dense and sparse labeling with multidimensional features for saliency detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 5, pp. 1130–1143, May 2018.

[5] R. Cong, J. Lei, H. Fu, Q. Huang, X. Cao, and N. Ling, "HSCS: Hierarchical sparsity based co-saliency detection for RGBD images," *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1660–1671, Jul. 2019.

[6] B. Lai and X. Gong, "Saliency guided end-to-end learning for weakly supervised object detection," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 2053–2059.

[7] Y. Wei *et al.*, "STC: A simple to complex framework for weakly-supervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2314–2320, Nov. 2017.

[8] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 185–198, Jan. 2010.

[9] R. Cong *et al.*, "An iterative co-saliency framework for RGBD images," *IEEE Trans. Cybern.*, vol. 49, no. 1, pp. 233–246, Nov. 2019.

[10] L. Zhu, X. Hu, C.-W. Fu, J. Qin, and P.-A. Heng, "Saliency-aware texture smoothing," *IEEE Trans. Vis. Comput. Graphics*, early access, Dec. 21, 2018, doi: 10.1109/TVCG.2018.2889055.

[11] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2015, pp. 597–606.

[12] R. Cong, J. Lei, H. Fu, F. Porikli, Q. Huang, and C. Hou, "Video saliency detection via sparsity-based reconstruction and propagation," *IEEE Trans. Image Process.*, vol. 28, no. 10, pp. 4819–4831, Oct. 2019.

[13] C. Lang, J. Feng, G. Liu, J. Tang, S. Yan, and J. Luo, "Improving bottom-up saliency detection by looking into neighbors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 6, pp. 1016–1028, Jun. 2013.

[14] S. Wang, M. Wang, S. Yang, and K. Zhang, "Salient region detection via discriminative dictionary learning and joint Bayesian inference," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 5, pp. 1116–1129, Jan. 2018.

[15] Y. Liu, J. Han, Q. Zhang, and L. Wang, "Salient object detection via two-stage graphs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 4, pp. 1023–1037, Apr. 2019.

[16] J. X. Zhao, J. J. Liu, D. P. Fan, J. Yang, and M. M. Cheng, "EGNet: Edge guidance network for salient object detection," in *Proc. ICCV*, 2019, pp. 8779–8788.

[17] 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, Aug. 2015.

[18] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *Proc. CVPR*, 2013, pp. 2083–2090.

[19] T. Liu *et al.*, "Learning to detect a salient object," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 353–367, Feb. 2011.

[20] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5455–5463.

[21] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1265–1274.

[22] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. S. Torr, "Deeply supervised salient object detection with short connections," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 815–828, Apr. 2019.

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

[24] Z. Deng *et al.*, "R^3net: Recurrent residual refinement network for saliency detection," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 684–690.

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

[26] X. Zhang, T. Wang, J. Qi, H. Lu, and G. Wang, "Progressive attention guided recurrent network for salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 714–722.

[27] J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, and J. Jiang, "A simple pooling-based design for real-time salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3917–3926.

[28] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "BASNet: Boundary-aware salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7479–7489.

[29] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8150–8159.

[30] M. Feng, H. Lu, and E. Ding, "Attentive feedback network for boundary-aware salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1623–1632.

[31] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2006, pp. 2169–2178.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Jan. 2015.

[33] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.

[34] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 733–740.

[35] A. Borji and L. Itti, "Exploiting local and global patch rarities for saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 478–485.

[36] V. Mahadevan and N. Vasconcelos, "Biologically inspired object tracking using center-surround saliency mechanisms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 3, pp. 541–554, Mar. 2013.

[37] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 1155–1162.

[38] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 3166–3173.

[39] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Proc. NIPS*, 2007, pp. 545–552.

[40] X. Li *et al.*, "DeepSaliency: Multi-task deep neural network model for salient object detection," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3919–3930, Aug. 2016.

[41] Z. Luo, A. Mishra, A. Achkar, J. Eichel, S. Li, and P.-M. Jodoin, "Non-local deep features for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6609–6617.

[42] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Saliency detection with recurrent fully convolutional networks," in *Proc. ECCV*, 2016, pp. 825–841.

[43] P. Zhang, D. Wang, H. Lu, H. Wang, and B. Yin, "Learning uncertain convolutional features for accurate saliency detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 212–221.

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

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

[46] W. Wang, J. Shen, X. Dong, and A. Borji, "Salient object detection driven by fixation prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1711–1720.

[47] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *Proc. ECCV*, 2018, pp. 234–250.

[48] L. Zhang, J. Dai, H. Lu, Y. He, and G. Wang, "A bi-directional message passing model for salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1741–1750.

[49] T. Wang *et al.*, "Detect globally, refine locally: A novel approach to saliency detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 3127–3135.

[50] L. Zhang, J. Zhang, Z. Lin, H. Lu, and Y. He, "CapSal: Leveraging captioning to boost semantics for salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6024–6033.

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

[52] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, Jul. 2001, pp. 416–423.

[53] L. Wang *et al.*, "Learning to detect salient objects with image-level supervision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 136–145.

[54] 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, Apr. 2018.

[55] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "DenseASPP for semantic segmentation in street scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 3684–3692.

[56] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[57] P. Wang *et al.*, "Understanding convolution for semantic segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1451–1460.

[58] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3907–3916.

[59] X. Li, F. Yang, H. Cheng, W. Liu, and D. Shen, "Contour knowledge transfer for salient object detection," in *Proc. ECCV*, 2018, pp. 355–370.

[60] Y. Zeng, H. Lu, L. Zhang, M. Feng, and A. Borji, "Learning to promote saliency detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1644–1653.

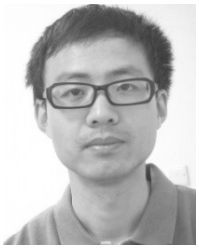
[61] T. Wang, A. Borji, L. Zhang, P. Zhang, and H. Lu, "A stagewise refinement model for detecting salient objects in images," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4019–4028.

[62] N. Liu and J. Han, "DHSNet: Deep hierarchical saliency network for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 678–686.

[63] G. Lee, Y.-W. Tai, and J. Kim, "Deep saliency with encoded low level distance map and high level features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 660–668.

[64] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3183–3192.

[65] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 110–119.
[66] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 248–255.
[67] G. Li and Y. Yu, "Visual saliency detection based on multiscale deep CNN features," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5012–5024, Nov. 2016.
[68] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, and A. Borji, "Enhanced-alignment measure for binary foreground map evaluation," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 698–704.
[69] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 1597–1604.
[70] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4548–4557.
[71] K. Fu, Q. Zhao, I. Yu-Hua Gu, and J. Yang, "Deepside: A general deep framework for salient object detection," *Neurocomputing*, vol. 356, pp. 69–82, Sep. 2019.
[72] 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 *Proc. ECCV*, 2018, pp. 186–202.
[73] X. Yang, H. Mei, K. Xu, X. Wei, B. Yin, and R. Lau, "Where is my mirror?" in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8809–8818.



Liansheng Wang (Member, IEEE) received the Ph.D. degree in computer science from The Chinese University of Hong Kong in 2012. He is currently an Associate Professor with the Department of Computer Science, Xiamen University, Xiamen, China. His research interest includes medical image processing and analysis.

Rongzhen Chen received the B.S. degree from Xiamen University, Xiamen, China, in 2017, where he is currently pursuing the master's degree with the Department of Computer Science. His research interests include medical image processing and machine learning.

Lei Zhu (Member, IEEE) received the Ph.D. degree from the Department of Computer Science and Engineering, The Chinese University of Hong Kong in 2017. He is currently working as a Post-Doctoral Fellow with The Chinese University of Hong Kong. His research interests include computer graphics, computer vision, medical image processing, and deep learning.

Haoran Xie (Senior Member, IEEE) received the Ph.D. degree in computer science from The City University of Hong Kong. He is currently an Associate Professor with the Department of Computing and Decision Sciences, Lingnan University, Hong Kong. He has successfully received more than 40 research grants; the total amount of these grants is more than HK$27 million. He has totally published 192 research publications, including 83 journal articles, 64 SCI/SSCI indexed, and 12 SCOPUS indexed. His research interests include artificial intelligence, big data, and educational technology. He is the Senior Member of ACM. He has received ten research awards, including the Golden Medal and the special award from the International Invention Innovation Competition in Canada, and the Silver Award from Geneva's Invention Expo. He has ranked as the world top 25 researchers in Artificial Intelligence in Education in Google Scholar. He has served as an Academic Editor of PLOS One and Education Research International, an Editorial Member of IJDET and JCE, a Guest Editor in special issues of journals, an Editor of six books, an Organization Committee Chair/Member of 57 conferences, and a Reviewer of 40 international journals.

Xiaomeng Li (Member, IEEE) received the Ph.D. degree from the Department of Computer Science and Engineering, The Chinese University of Hong Kong. She is currently a Post-Doctoral Researcher with The Chinese University of Hong Kong. Her research interests include computer vision and deep learning.