SDCNet: Size Divide and Conquer Network for Salient Object Detection

Senbo Yan[1][0000−0002−5051−0506], Xiaowen Song[1][0000−0001−6386−9836], and Chuer Yu[2][0000−0003−4701−4787]

[1] State Key Laboratory of Fluid Power and Mechatronic Systems, Zhejiang University, Hangzhou 310027, China.
{3140100833,songxw}@zju.edu.cn
[2] College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China.
11721080@zju.edu.cn

Abstract. The fully convolutional neural network (FCN) based methods achieve great performances in salient object detection (SOD). However, most existing methods have difficulty in detecting small or large objects. To solve this problem, we propose Size Divide and Conquer Network (SDCNet) which learning the features of salient objects of different sizes separately for better detection. Specifically, SDCNet contains two main aspects: (1)We calculate the proportion of objects in the image (with the ground truth of pixel-level) and train a size inference module to predict the size of salient objects. (2)We propose a novel Multi-channel Size Divide Module (MSDM) to learning the features of salient objects with different sizes, respectively. In detail, we employ MSDM following each block of the backbone network and use different channels to extract features of salient objects within different size range at various resolutions. Unlike coupling additional features, we encode the network based on the idea of divide and conquer for different data distributions, and learn the features of salient objects of different sizes specifically. The experimental results show that SDCNet outperforms 14 state-of-the-art methods on five benchmark datasets without using other auxiliary techniques.

1 Introduction

Salient object detection (SOD), which aims to find the distinctive objects in the image, plays an important role in many computer vision tasks, such as weakly supervised semantic segmentation [1], object recognition [2], image parsing [3] and image retrieval [4]. Besides, there is a lot of work focused on RGB-D salient object detection [5–8] and video salient object detection [9, 10]. One main problem of SOD is that the salient objects in different images have extremely different sizes. As shown in Figure 1, the size of salient objects under the same dataset varies greatly. We define the pixel-wise ratio between the salient objects and the entire image as salient object proportion (SOP). We show the SOP distribution of 10 benchmark datasets in Table 1 and Figure 2. It is observed that 25% of

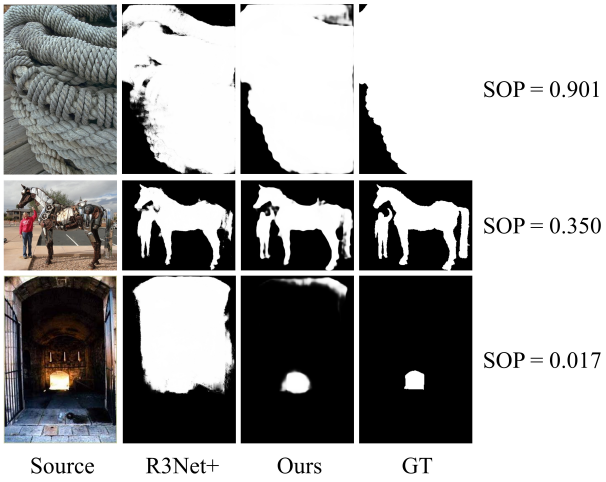

Fig. 1. Saliency detection results for salient objects with different size by different methods. We use salient object proportion (SOP) to indicates the proportion of objects in the whole image.

images have the SOP less than 0.1, while 10% of images have the SOP larger than 0.4. The size difference of salient objects is ubiquitous in SOD datasets.

Some FPN-based complex multi-level feature fusion methods [11–13] try to alleviate the perplex caused by huge size deviation. However, multiscale feature fusion methods generally ignore the difference in data distribution determined by the huge size deviation, which leads to differences in the features that need to be learned.These methods have been proved cannot completely solve the problem of the size difference. [14] proved that performances of SOD methods generally decreased in small objects (SOP between 0 to 0.1) or large objects (SOP above 0.5). Our method is based on a basic fact: for a network with the same structure, if only small-size (or large-size) objects are used for training, the model performance in small-size (or large-size) objects detection will be better than using the entire dataset for training. Moreover, the size difference of the salient objects has an intuitive impact. For example, the detection of small objects depends more on local information, while large objects contain more global semantic information. Existing SOD methods ignore the size difference of salient objects. We argue that divide and conquer objects of different sizes can lead to a more robust model and better performances.

In this paper, we regard the size information as a beneficial supplement to the salient object information and propose a novel method to divide and conquer salient objects of different sizes. Firstly, we establish an FPN-based side output architecture to realize the fusion of features at high and low levels. The only reason we employ multi-resolution fusion is to make a fair comparison with SOTA methods that generally uses feature fusion to improve performance. Secondly, we

Table 1. Distribution of salient object proportion (SOP) in 10 benchmark datasets without non-saliency images. The size range is divided into five categories according to SOP. "-10%" means SOP between 0 to 0.1.

Dataset	-10%	10%-20%	20%-30%	30%-40%	40%-	Total
MSRA10K	1020	3646	3060	1756	518	10000
ECSSD	154	326	244	130	136	1000
DUT-O	2307	1387	818	418	238	5168
DUTS-TR	1239	2656	2553	1994	2111	10553
DUTS-TE	2299	1506	626	234	354	5019
HKU-IS	983	1580	1179	510	195	4447
PASCAL-S	204	187	168	139	146	844
SED2	34	27	18	5	16	100
SOD	54	70	68	37	69	298
THUR-15K	2616	2083	779	429	326	6233
Total	11681	14036	9920	5948	4477	48462

obtain the size inference of salient objects through a Size Inference Module (SIM) which shares the same backbone with SDCNet. SIM generates a binarized rough saliency inference and the predicted size range of salient objects is obtained by calculating SOP. As shown in Table 1, we classify the size range into five categories according to the SOP (0-10%, 10%-20%, 20%-30, 30%-40% and above 40%). In the side output structure, we add MSDM in the process of feature fusion up-to-down. MSDM divides feature maps of each side layer into size-independent stream and size-dependent stream. As shown in Figure 4, we put the size-independent stream into a common convolutional layer and put size-dependent stream into multi-channel convolutional layers. Each channel of multi-channel convolutional layers corresponds to a specific size range. We integrate size-independent features with complementary size-dependent features.

Finally, we refer to [15] and add a one-to-one guidance module based on low-level feature maps to enhance the network sensitivity to small-size objects. In summary, the main contributions of this paper include three folds:

1. We propose a novel network design method that divides and conquers different data distributions. MSDM can learn the features of salient objects in different size ranges separately. This network design based on data characteristics is meaningful.
2. We provide an effective idea of solving the huge size deviation between salient objects, which significantly improves the accuracy of saliency maps.
3. We compare the proposed method with 14 state-of-the-art methods on five benchmark datasets. Our method achieves better performances over three evaluation metrics without pre-processing and post-processing.

2 Related work

In the last few years, lots of methods have been proposed to detect salient objects in images. The early methods mainly used hand-craft low-level features,

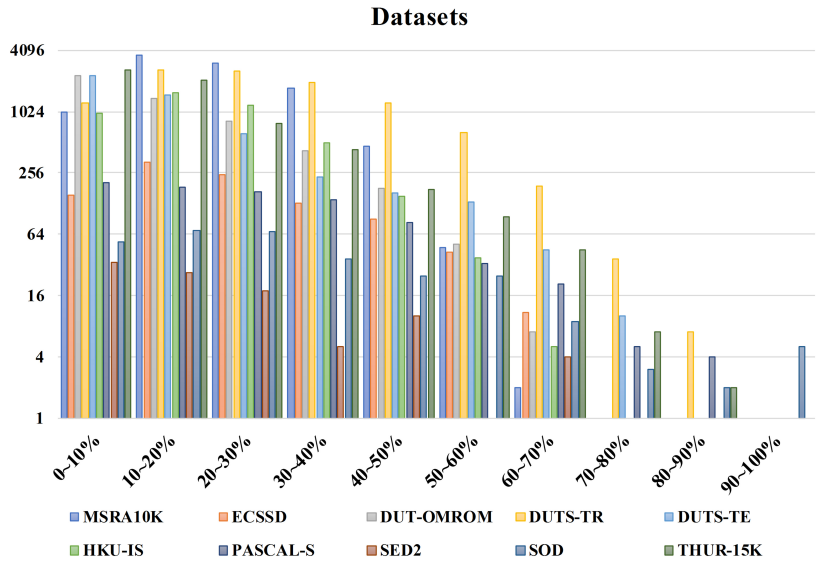

Fig. 2. Size distribution in 10 benchmark datasets

such as color contrast [16], local contrast [17], and global contrast [18]. In recent years, FCN-based methods [19] have completely surpassed traditional methods. Most of FCN-based methods are devoted to better integrate multi-level features or enhance the utilization of important features so as to improve the performance [11,15,20–23]. [13] improves the traditional progressive structure of FPN. It aggregates multi-scale features from different layers and distributes the aggregated features to all the involved layers to gain access to richer context. For fair comparison, we employ the original FPN architecture in our network.

[11] designed a HED-based side output structure which using the short connection to integrate the low-level features in the shallow side layer and high-level features in the deep side layer to improve the effect of saliency prediction. Instead of using layer skipped dense short connections, we retain multi-channel concatenation layer by layer as our basic architecture.

Recently, methods based on edge feature enhancement have been widely studied. [24] proposed to learn the local context information of each spatial location to refine the boundaries. [15] proposed to use the complementarity of edge information and saliency information to enhance the accuracy of boundary and help to locate salient objects. [20] fused the boundary and internal features of salient objects through selectivity or invariance mechanism. Because edge information has a significant effect in improving the pixel-wise edge accuracy of salient objects, a lot of edge information enhancement methods have been proposed [15,20,25,26]. [21] have also used edge information as an important way to enhance the performance of network. These methods utilize additional edge

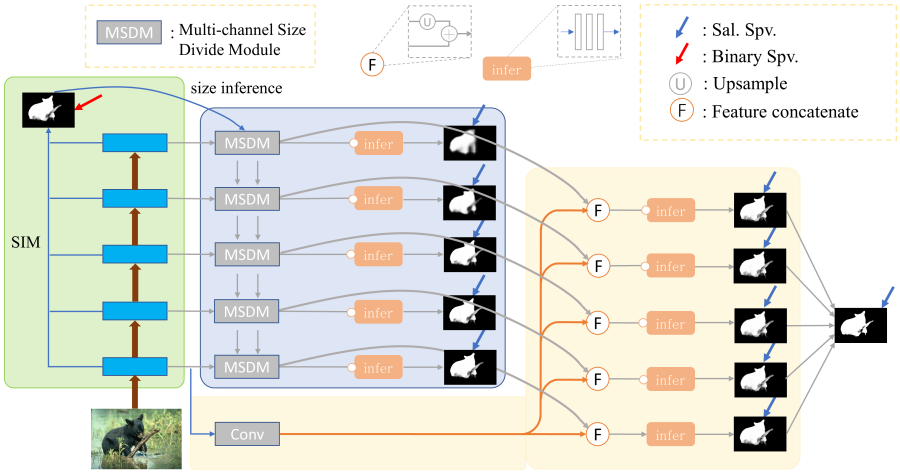

Fig. 3. Main pipeline of the proposed method. We integrate MSDM into improved FPN structure. Two parallel arrows between MSDM indicate the flow of size-independent and size-dependent features respectively. The green block on the left represents the size inference module (SIM) and yellow block represents the size guide module (SGM).

information to solve the issue of rough edge prediction of salient objects in previous methods. These researches inspire us to explore the usage of size information of salient objects, which is equally important and facing difficulties currently. However, these two problems are opposite. Edge information is the common feature of salient objects, while size information emphasizes the feature difference between salient objects. Moreover, edge detection can be easily integrated into saliency detection tasks by multi-learning. Correlation of size categories and saliency detection by multi-learning directly does not make sense.Therefore, we combined the divide-and-conquer algorithm and designed MSDM to extract the corresponding features of salient objects of different sizes. It uses the size categories as high-dimensional information, activates different convolution channels to extract the features of salient objects of different sizes, and effectively solves the problem of the confusion of salient object sizes.

3 Methodology

The performance decline of SOD methods at small objects and large objects indicates that we need different features in detecting salient objects with huge size difference. Ignoring the size difference of saliency objects will suppress learning of features that are related to specific size. Accordingly, we design a MSDM for our improved FPN structure. The overall structure of SDCNet is shown in Figure 3.

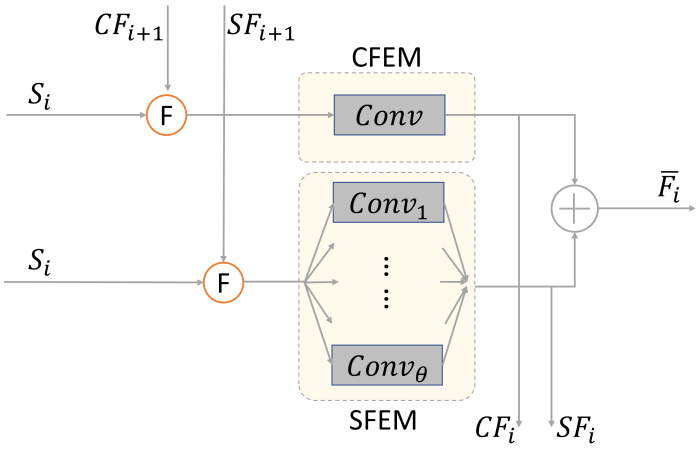

Fig. 4. Structure of MSDM. We use Common Feature Extract Module (CFEM) to get size-independent features and Size Feature Extract Module (SFEM) to get size-dependent features. We activate different convolutional channel in SFEM according to the size inference θ. The details are introduced in the Sec. 3.2.

3.1 Overall Pipeline

Our proposed network is independent of the specific backbone network. In practice, we use ResNet [27] as backbone for feature extraction. We remove the last pooling layer and fully connection layer and get 5 side features F_1, F_2, F_3, F_4, F_5 which including 64, 256, 512, 1024 and 2048 channels respectively as our side output path S_1, S_2, S_3, S_4, S_5. The side outputs of these different layers contain low-level details and high-level semantic information respectively. We employ Atrous Spatial Pyramid Pooling (ASPP) in processing two high-level feature maps F_4, F_5 to expand receptive field of convolutional layer. We use SIM to provide size inference. MSDM is added to each layer of the up-to-down structure to replace the simple feature fusion module. The main function of MSDM is to activate different convolution channels through the high-dimensional size information provided by SIM to learn the feature of salient objects of different sizes. It integrates basic function of upsample and concatenate as well. Finally, we add an one-to-one guidance module with low-level features to retain more information of small size salient objects. Our network is end-to-end, training and inference are both one-stage. The details of the convolutional layers could be found in Table 2.

3.2 Multi-channel Size Divide Module

The MSDM is mainly composed of the common saliency feature extraction module (CFEM) and size-dependent feature extraction module (SFEM). The structure of MSDM can be found in Figure 4. CEFM is a single set of convolutional

Table 2. Details of kernels in SDCNet. We use ResNet50 for example. "3×3, 256" means the keneral size is 3×3 and channel number is 256. Each Conv layer follow with BN and PRelu and each side-layer share the same setting.

SIM	Backbone	CFEM	SFEM	SGM
	Conv1_1			
3×3,256	Conv2_3	3×3,256	(3×3,256)	3×3,256
3×3,256	Conv3_4	3×3,256	(3×3,256)×5	3×3,256
1×1,128	Conv4_6	1×1,128	(1×1,128)	1×1,128
	Conv5_3			

layers, while SEFM is a combination of multiple sets of parallel convolutional layers. In CFEM, we extract size-independent features through a common convolutional layer. CFEM remains active for salient objects of all sizes, that is, it learns common features that are not related to size differences. In SFEM, we activate different convolutional set to independently extract size-dependent features according to the specific size categories. We integrate these two complementary features to generate saliency maps in each side path S_1, S_2, S_3, S_4, S_5 up-to-down. The size-independent and size-dependent features of each layer are denoted as follows:

$$CF_i = f_{conv}^{(i)}(Cat(F_i, Up(CF_{(i+1)}; F_i))), \tag{1}$$

$$SF_i = f_{(conv,\theta)}^{(i)}(Cat(F_i, Up(SF_{(i+1)}; F_i)), \theta), \tag{2}$$

where CF_i represents size-independent feature maps, SF_i represents size-dependent feature maps. $Up(*; F_i)$ means up-sampling * to the same size of F_i through bilinear interpolation. $Cat(A, B)$ means concatenation of feature maps A and B. $f_{conv}^{(i)}$ represents CFEM which is composed by three convolutional layers and nonlinear activation function. The structure of $f_{(conv,\theta)}^{(i)}$ is composed by several parallel $f_{conv}^{(i)}$, and we activate one of them for each image according to size inference θ. $f_{(conv,\theta)}^{(i)}$is applied to extract size-dependent features. θ represents size inference, which provided by SIM. Specific characteristics of θ are as follows:

$$\theta = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |S(x, y)|, \tag{3}$$

where W and H represent the width and height of the image respectively, and $S(x, y)$ represents the binarized pixel-wise value of saliency maps. In practice, size inference θ is divided into five categories according to the value. More details are shown in Table 1.

Deep supervision is applied on each side path S_i. We integrate CF_i and SF_i to make saliency inference P_i and impose the supervision signal each layer. The specific expression is as follows:

$$\begin{cases} \bar{F}_i = Cat(CF_i, SF_i, P_{(i+1)}), & 1 \le i \le 4, \\ \bar{F}_i = Cat(CF_i, SF_i), & i = 5, \end{cases} \tag{4}$$

$$P_i = Sig(Up(Pred_{conv}^{(i)}(F_i); S_{img})), \tag{5}$$

where P_i denotes the salieny prediction of i-th side layer. $\bar{F}_i$ represents feature maps aggregated by size-independent features and size-dependent features. Similar to $f_{conv}^{(i)}$, $Pred_{conv}^{(i)}$ is a series of convolutional layers for salient object prediction. S_{img} denotes the size of source image. $Up(*; S_{img})$ means up-sampling the saliency prediction * to the same size as source image. $Sig(*)$ means sigmoid function.

To realize the supervision in various resolutions, we design loss function for each side output based on the cross-entropy loss function. The formula is as follows:

$$\begin{aligned} \mathbb{L}_i(G, P_i) = -\frac{w_i}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} [g^{xy} \log p_i^{xy} \\ + (1 - g^{xy}) \log(1 - p_i^{xy})], \qquad i \in [1, 5], \end{aligned} \tag{6}$$

where G denotes the input image GT. g^{xy} and p_i^{xy} represent the pixel-wise value of GT and the normalized saliency prediction. w_i is used to represent the weight of loss function of each layer and the value is 1. For the MSDM, our total loss function can be expressed as:

$$\Theta = \sum_{i=1}^{5} \mathbb{L}_i(G, P_i). \tag{7}$$

3.3 Size Inference Module

As shown in Figure 3, We generate binary predictions of salient objects through multi-level feature fusion. SIM share the same backbone with the main network and the loss function of SIM is similar to the loss function in Sec 3.2. Unlike the usual non-binary saliency inference, we get a tensor of size (2, H, W). Channel 1 represents the salient objects and Channel 2 represents the background. We directly generate a binarized inference to conveniently calculate the SOP of the images pixel by pixel. For example, for an input data with a batchsize of 8, we separately infer the salient object area of each image, and calculate the SOP. According to the SOP, we generates a vector of length 8, which represents the predicted size category of each image. This size category is determined by the five size ranges shown in Table 1. This is a rough size estimate, but they all belong to the same category within a certain range, so the size category is usually accurate. The accuracy rate of the inference of the salient object size categories on different data sets is shown in Table 6. Finally, we employ this size category inference as high-dimensional information to guide the activation of different channels in MSDM.

3.4 Size Guidance Module

Since small objects suffer more information loss during the down-sampling, we use side path $\bar{F}_0$ of the shallow layer as the guidance layer to provide more low-level features. The guidance layer $\bar{F}_0$ and the sub-side layer $\bar{F}_i^*$ can be expressed

as follows respectively:

$$\bar{F}_0 = f_{conv}^{(1)}(F_1), \tag{8}$$

$$\bar{F}_i^* = f_{conv}^{*(i)}(Up(\bar{F}_i; \bar{F}_1) + \bar{F}_0). \tag{9}$$

similar to MSDM, we use a series of convolutional layers $f_{conv}^{*(i)}$ to generate aggregated feature maps. We employ an inference module to generate the second round of saliency predictions. The setting of inference module and the loss function are the same as those described in Sec. 3.2.

4 Experiments

4.1 Experiment Setup

Implementation Details. We implement the model with PyTorch 0.4.0 on Titan Xp.We use ResNet [27] and ResNeXt [28] as the backbone of our network, respectively. We use the SGD algorithm to optimize our model, where the batch size is 8, momentum is 0.9, weight decays is 5e-4. We set the initial learning rate to 1e-4 and adopt polynomial attenuation with the power of 0.9. We iterate our model for 30,000 times and do not set the validation set during training. We use the fused prediction maps of side output as the final saliency map.

Datasets. We have evaluated the proposed method on five benchmark datasets: DUTS-TE [29], ECSSD [30], PASCAL-S [32], HKU-IS [33], DUT-OMRON [34]. DUTS [29] is a large SOD dataset containing 10553 images for training (DUT-TR) and 5019 images for testing (DUT-TE). Most images are challenging with various locations and scales as well as complex backgrounds. ECSSD [30] contains 1000 images with complex structures and obvious semantically meaningful objects. PASCAL-S [32] is derived from PASCAL VOC 2010 segmentation dataset and contains 850 natural images. HKU-IS [33] contains 4447 images and many of which have multiple disconnected salient objects or salient objects that touch image boundaries. DUT-OMRON [34] contains 5168 high-quality but challenging images. These images are chosen from more than 140,000 natural images, each of which contains one or more salient objects and relatively complex backgrounds.

4.2 Evaluation Metrics

We adopt mean absolute error (MAE), Max F-measure (F_β^{Max}) [36], and a structure-based metric, S-measure [37], as our evaluation metrics. MAE reflects the average pixel-wise absolute difference between the normalized saliency maps and GT. MAE can be calculated by:

$$MAE = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |P(x,y) - G(x,y)| \tag{10}$$

where W and H represent the width and height of images. $P(x, y)$ and $G(x, y)$ denote the saliency map and GT, respectively.

F-measure is a harmonic mean of average precision and average recall. We compute the F-measure by:

$$F_\beta = \frac{(1+\beta^2)Precision \times Recall}{\beta^2 Precision + Recall} \tag{11}$$

we set $\beta^2 = 0.3$ to weigh more on precision than recall as suggested in [16]. Precision denotes the ratio of detected salient pixels in the predicted saliency map. Recall denotes the ratio of detected salient pixels in the GT. We normalize the predicted saliency maps into the range of [0, 255] and binarize the saliency maps with a threshold from 0 to 255. By comparing the binary maps with GT, we can get precision-recall pairs at each threshold and then evaluate the maximum F-measure from all precision-recall pairs as F_β^{Max}.

S-measure is proposed by [14], and it can be used to evaluate the structural information of non-binary foreground maps. This measurement is calculated by evaluating the region-aware and object-aware structural similarity between the saliency maps and GT.

4.3 Comparisons with State-of-the-arts

In this section, we compare the proposed method with 14 state-of-the-art methods, including EGNet [15], BANet [20], RAS [39], RADF [40], R3Net [12], PiACNet [41], PAGRN [22], DGRL [24], BDMPM [42], SRM [43], NLDF [44], DSS [11], Amulet [45] and UCF [46]. All of these methods are proposed in the last three years. Saliency maps of the above methods are produced by running source codes with original implementation details or directly provided by the authors. We evaluating the saliency maps both in the code provide by [11] and by ourselves to guarantee the reliability of the results.

F-measure, MAE, and S-measure. We compared with 14 state-of-the-art saliency detection methods on five datasets. The comparison results are shown in Table 3. We can see that our method significantly outperforms other methods across all of the six benchmark datasets in MAE. Specifically, our method reduce MAE by 17.9%, 10.8%, 25.6%, 6.5% and 15.1% on DUTS-TE [29], ECSSD [30], PASCAL-S [32], HKU-IS [33] and DUT-OMRON [34] datasets, respectively. For the metrics where we get top two or three, we are only slightly behind the best edge-guide method. In fact, without using edge information, we achieved state-of-the-art performance comparable to the best model combining edge information.

Visual comparison. We show some visualization results in Figure 5. Those pictures have different SOP: 0.866, 0.570, 0.281, 0.167, 0.134, 0.0864, 0.042, 0.034, from up-to-down. It is obvious that our method has consistent performance for salient objects of different sizes. A significant advantage of our method is that

Table 3. Comparison of 14 state-of-the-arts and the proposed method on Max F, MAE, and S-measure over five benchmark datasets. ↑ & ↓ denote higher better and lower better respectively. * means the results are post-processed by dense conditional random field (CRF) [38]. The best three results are marked in red, and green, blue. Our method achieves the state-of-the-art on these five benchmark datasets under three evaluation metrics.

Method	DUTS-TE			ECSSD			PASCAL-S			HKU-IS			DUT-O		
	Fm↑	MAE↓	S↑	Fm↑	MAE↓	S↑	Fm↑	MAE↓	S↑	Fm↑	MAE↓	S↑	Fm↑	MAE↓	S↑
VGG-based															
UCF	0.772	0.111	0.782	0.903	0.069	0.884	0.816	0.116	0.806	0.888	0.062	0.875	0.730	0.120	0.760
Amulet	0.778	0.084	0.804	0.915	0.059	0.894	0.830	0.100	0.818	0.897	0.051	0.886	0.743	0.098	0.781
NLDF	0.816	0.065	0.805	0.905	0.063	0.875	0.824	0.098	0.805	0.902	0.048	0.879	0.753	0.080	0.770
RAS	0.831	0.059	0.839	0.921	0.056	0.893	0.831	0.101	0.799	0.913	0.045	0.887	0.786	0.062	0.814
DSS*	0.825	0.061	0.812	0.921	0.052	0.882	0.833	0.093	0.799	0.916	0.040	0.878	0.781	0.063	0.790
RADF	-	-	-	0.923	0.049	0.894	0.832	0.097	0.802	0.914	0.039	0.889	0.791	0.062	0.815
PAGRN	0.854	0.055	0.839	0.927	0.061	0.889	0.849	0.089	0.822	0.918	0.048	0.887	0.771	0.071	0.775
BDMPM	0.854	0.048	0.862	0.930	0.045	0.911	0.858	0.074	0.844	0.922	0.039	0.907	0.793	0.064	0.809
ResNet-based															
SRM	0.826	0.058	0.836	0.917	0.054	0.895	0.840	0.084	0.834	0.906	0.046	0.887	0.769	0.069	0.798
DGRL	0.828	0.050	0.842	0.922	0.041	0.903	0.849	0.072	0.836	0.910	0.036	0.895	0.774	0.062	0.806
PiCA*	0.871	0.040	0.863	0.940	0.035	0.916	0.870	0.064	0.846	0.929	0.031	0.905	0.828	0.054	0.826
BANet	0.872	0.039	0.879	0.943	0.035	0.924	0.866	0.070	0.852	0.931	0.032	0.913	0.803	0.059	0.832
EGNet	0.893	0.039	0.887	0.943	0.037	0.925	0.869	0.074	0.852	0.937	0.031	0.918	0.842	0.053	0.841
Ours	0.888	0.032	0.884	0.943	0.033	0.922	0.871	0.055	0.863	0.933	0.029	0.914	0.835	0.045	0.839
ResNeXt-based															
R3Net*	0.879	0.037	0.873	0.938	0.036	0.913	0.867	0.063	0.851	0.930	0.029	0.910	0.818	0.057	0.819
Ours	0.896	0.030	0.890	0.948	0.029	0.926	0.880	0.058	0.872	0.937	0.026	0.918	0.845	0.041	0.844

we can locate the main area of salient objects more accurately. As shown in row 6 and row 8, when the salient objects share the same attributes with background or the background is relatively complex, our method can accurately segment the main objects and dropout the extra parts. Another advantage of our approach is that we retain more detail. In row 4, we preserve more details compare with other methods. It proves that the shallow feature guidance layer is effective for retaining detail information. In addition to achieving marvelous performance on small objects, we achieve impressive results on large objects as well (row 1-3). These observations indicate the size information of salient objects is crucial for identifying the salient objects and improving the robustness of the SOD methods at multiple scales.

4.4 Ablation Analysis

To explore the effectiveness of different components of the proposed method, we conduct experiments on five benchmark datasets to compare the performance

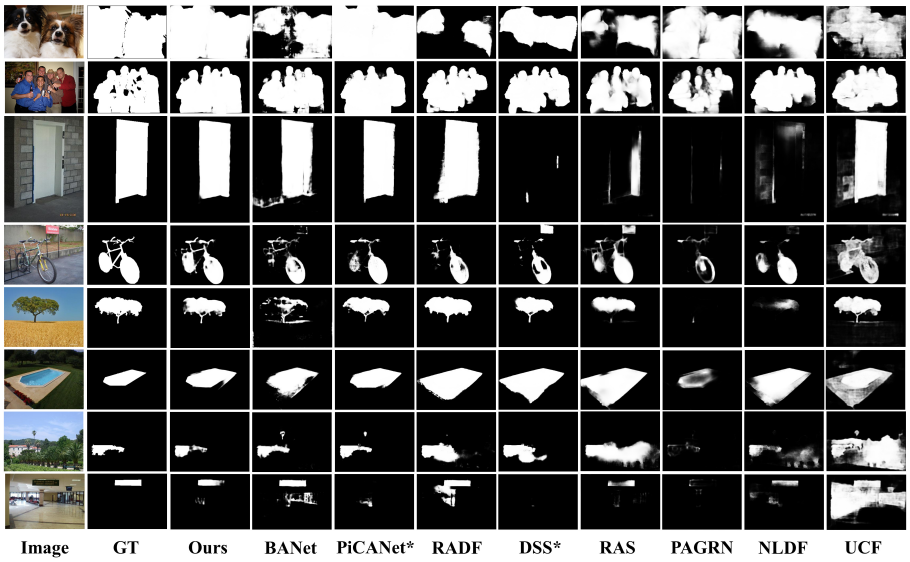

Fig. 5. Qualitative comparisons with 8 state-of-the-art methods. We arrange those images from high to low SOP up-to-down. * means the results are post-processed by dense conditional random field (CRF).

variations of our methods with different experimental settings over the DUTS-TE [29], ECSSD [30], PASCAL-S [32] and DUT-OMRON [34]. Test results of different settings are shown in Table 4.

Effectiveness of MSDM. We explore the effectiveness of MSDM in this subsection. As shown in Table 4, CFEM denotes only remain CFEM in MSDM. SFEM denotes remain SFEM in MSDM. For better comparison, we kept other settings the same. The comparison between the first and third columns of Table 4 proves the effectiveness of SFEM and MSDM. It means that the divide-and-conquer module is effective in separately learning the features of salient objects of different sizes. By comparing the second and third columns we can find that retaining a common convolutional layer can better learning size-independent features and improve the network in a complementary way. MSDM+edge verifies the effectiveness of edge information. SDCNet achieved higher performance by combining edge information. Running time in CFEM, SFEM and CFEM+SFEM is 6.71, 6.58 and 6.06 FPS in Titan Xp with input size 300×300. In addition, MSAM can easily integrate into other lightweight networks.

Improvement on small and big object detection. To demonstrate the superiority of SDCNet in the detection of small and big salient objects, we compared the performance difference between the baseline network (the same as CFEM in Table 4) and SDCNet in the detection of small objects and large

Table 4. Ablation analyses on four datasets. CFEM, SFEM, MSDM are introduced in Sec. 3.2. MSAM+edge means add edge supervision to $\bar{F}_0$.

Model		CFEM	SFEM	MSDM (CFEM+SFEM)	MSDM+edge
DUTS-TE	MaxF↑	0.875	0.890	0.896	0.898
	MAE↓	0.042	0.033	0.030	0.028
ECSSD	MaxF↑	0.937	0.944	0.948	0.951
	MAE↓	0.035	0.031	0.029	0.027
PASCAL-S	MaxF↑	0.868	0.878	0.880	0.879
	MAE↓	0.067	0.058	0.058	0.058
DUT-O	MaxF↑	0.818	0.837	0.845	0.847
	MAE↓	0.063	0.045	0.041	0.040

Table 5. Performance improvement of SDCNet in small and big object detection. Small objects means SOP is less than 10%. Large objects refers to SOP more than 40%. We use the complete dataset to train the baseline and SDCNet, and test on large and small objects separately. Specific Dataset means training with a dataset of the same size category as the test set.

Model			Baseline	SDCNet	Specific Dataset
Small Objects	DUTS-TE	MaxF↑	0.823	0.850	0.872
		MAE↓	0.038	0.027	0.015
	PASCAL-S	MaxF↑	0.766	0.794	0.820
		MAE↓	0.062	0.045	0.021
Big Objects	DUTS-TE	MaxF↑	0.957	0.960	0.962
		MAE↓	0.056	0.052	0.048
	PASCAL-S	MaxF↑	0.923	0.933	0.932
		MAE↓	0.088	0.086	0.080

objects. The specific performance is shown in Table 5. SDCNet outperforms baseline in both small objects detection and large objects detection, while they sharing the same structure except SFEM. It demonstrates the superiority of the divide and conquer network in fitting actual data distribution. However, it still has performance differences compared to networks trained with specific data. The third column of Table 5 shows the best performance that can be achieved on small (or large) salient objects by dividing and conquering without changing other network structures.

Effectiveness of SIM The performance of SIM determines whether we can accurately divide the image into the corresponding channel. We explore the effectiveness of SIM in this subsection. As shown in Tabel 6, Classification Network denotes train an individual ResNeXt101 to infer the size category of salient objects. The results of the comparison show that the SIM module has a better accuracy of size inference than the independent size classification network. It indicates that it is more effective to calculate the size range of salient objects pixel by pixel than direct classification the size categories. The inference accuracy of SIM is about 80% to 85%. This does not seem completely satisfactory. But in fact, for those significant objects whose size range is wrongly estimated, the deviation is usually not large. The difference in features of salient objects

Table 6. Accuracy of the size inference on benchmark datasets. The details are introduced in Sec. 4.4.

Dataset		ECSSD	DUT-TE	HKU-IS	DUT-O
Classification Network (ResNeXt101)	acc(%)	74.8	71.7	72	69.6
SIM	acc(%)	85.4	83.6	84.7	80.2

in the size range of 30-40% and 40-50% is obviously smaller than that of the salient objects in the size range of 0-10%, so misclassification usually does not lead to worse performance. For those salient objects with large size inference deviation, the accuracy of labeling may be a more important reason. Moreover, the improvement space of SIM shows that SDCNet still has rich potential.

5 Conclusion

In this paper, we view size information as an important supplement of current SOD methods. We explored the application of divide-and-conquer networks in solving salient object detection with significant size differences. First, we counted the size distribution of salient objects in the benchmark datasets and trained a SIM to perform size inference using a pixel-by-pixel calculation. Second, we use an up-to-down multi-scale feature fusion network as the basic structure. We designed an MSDM, which activates different channels according to the size inference obtained by SIM, and learned the features of salient objects of different sizes. Finally, we utilize the low-level feature maps as one-to-one guidance to retain more information about small salient objects. Experimental results show that our method has a significant improvement in the detection performance of small-sized objects. Our method obtains state-of-the-art performance in five benchmark datasets under three evaluation metrics. Without using edge features, SDCNet can get results comparable to models that combine edge information. This impressive performance denotes the great effectiveness of our method. Furthermore, our method provides an original idea on how to overcome the inherent feature differences between task data and better solve the problems.

References

1. Hou, Q., Jiang, P., Wei, Y., Cheng, M.M.: Self-erasing network for integral object attention. In: Advances in Neural Information Processing Systems. (2018) 549–559
2. Ren, Z., Gao, S., Chia, L.T., Tsang, I.W.H.: Region-based saliency detection and its application in object recognition. IEEE Transactions on Circuits and Systems for Video Technology **24** (2013) 769–779
3. Lai, B., Gong, X.: Saliency guided dictionary learning for weakly-supervised image parsing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 3630–3639
4. He, J., Feng, J., Liu, X., Cheng, T., Lin, T.H., Chung, H., Chang, S.F.: Mobile product search with bag of hash bits and boundary reranking. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE (2012) 3005–3012

5. Piao, Y., Ji, W., Li, J., Zhang, M., Lu, H.: Depth-induced multi-scale recurrent attention network for saliency detection. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 7254–7263
6. Zhao, J.X., Cao, Y., Fan, D.P., Cheng, M.M., Li, X.Y., Zhang, L.: Contrast prior and fluid pyramid integration for rgbd salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 3927–3936
7. Chen, H., Li, Y.: Three-stream attention-aware network for rgb-d salient object detection. IEEE Transactions on Image Processing **28** (2019) 2825–2835
8. Chen, H., Li, Y., Su, D.: Multi-modal fusion network with multi-scale multi-path and cross-modal interactions for rgb-d salient object detection. Pattern Recognition **86** (2019) 376–385
9. Fan, D.P., Wang, W., Cheng, M.M., Shen, J.: Shifting more attention to video salient object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2019) 8554–8564
10. Wang, Z., Xu, J., Liu, L., Zhu, F., Shao, L.: Ranet: Ranking attention network for fast video object segmentation. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 3978–3987
11. Hou, Q., Cheng, M.M., Hu, X., Borji, A., Tu, Z., Torr, P.H.: Deeply supervised salient object detection with short connections. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 3203–3212
12. Deng, Z., Hu, X., Zhu, L., Xu, X., Qin, J., Han, G., Heng, P.A.: R3net: Recurrent residual refinement network for saliency detection. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence, AAAI Press (2018) 684–690
13. Li, Z., Lang, C., Liew, J., Hou, Q., Li, Y., Feng, J.: Cross-layer Feature Pyramid Network for Salient Object Detection. arXiv e-prints (2020) arXiv:2002.10864
14. Fan, D.P., Cheng, M.M., Liu, J.J., Gao, S.H., Hou, Q., Borji, A.: Salient objects in clutter: Bringing salient object detection to the foreground. In: Proceedings of the European conference on computer vision (ECCV). (2018) 186–202
15. Zhao, J.X., Liu, J.J., Fan, D.P., Cao, Y., Yang, J., Cheng, M.M.: Egnet: Edge guidance network for salient object detection. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 8779–8788
16. Achanta, R., Hemami, S., Estrada, F., Susstrunk, S.: Frequency-tuned salient region detection. In: 2009 IEEE conference on computer vision and pattern recognition, IEEE (2009) 1597–1604
17. Klein, D.A., Frintrop, S.: Center-surround divergence of feature statistics for salient object detection. In: 2011 International Conference on Computer Vision, IEEE (2011) 2214–2219
18. Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H., Hu, S.M.: Global contrast based salient region detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **37** (2014) 569–582
19. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2015) 3431–3440
20. Su, J., Li, J., Zhang, Y., Xia, C., Tian, Y.: Selectivity or invariance: Boundary-aware salient object detection. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 3799–3808
21. Liu, J.J., Hou, Q., Cheng, M.M., Feng, J., Jiang, J.: A simple pooling-based design for real-time salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 3917–3926

22. Zhang, X., Wang, T., Qi, J., Lu, H., Wang, G.: Progressive attention guided recurrent network for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 714–722
23. Wang, W., Shen, J., Cheng, M.M., Shao, L.: An iterative and cooperative top-down and bottom-up inference network for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 5968–5977
24. Wang, T., Zhang, L., Wang, S., Lu, H., Yang, G., Ruan, X., Borji, A.: Detect globally, refine locally: A novel approach to saliency detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 3127–3135
25. Feng, M., Lu, H., Ding, E.: Attentive feedback network for boundary-aware salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 1623–1632
26. Qin, X., Zhang, Z., Huang, C., Gao, C., Dehghan, M., Jagersand, M.: Basnet: Boundary-aware salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 7479–7489
27. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 770–778
28. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 1492–1500
29. Wang, L., Lu, H., Wang, Y., Feng, M., Wang, D., Yin, B., Ruan, X.: Learning to detect salient objects with image-level supervision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 136–145
30. Yan, Q., Xu, L., Shi, J., Jia, J.: Hierarchical saliency detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2013) 1155–1162
31. Li, Y., Hou, X., Koch, C., Rehg, J.M., Yuille, A.L.: The secrets of salient object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2014) 280–287
32. Li, G., Yu, Y.: Visual saliency based on multiscale deep features. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2015) 5455–5463
33. Yang, C., Zhang, L., Lu, H., Ruan, X., Yang, M.H.: Saliency detection via graph-based manifold ranking. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2013) 3166–3173
34. Margolin, R., Zelnik-Manor, L., Tal, A.: How to evaluate foreground maps? In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2014) 248–255
35. Fan, D.P., Cheng, M.M., Liu, Y., Li, T., Borji, A.: Structure-measure: A new way to evaluate foreground maps. In: Proceedings of the IEEE international conference on computer vision. (2017) 4548–4557
36. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. In: Advances in neural information processing systems. (2011) 109–117
37. Chen, S., Tan, X., Wang, B., Hu, X.: Reverse attention for salient object detection. In: Proceedings of the European Conference on Computer Vision (ECCV). (2018) 234–250
38. Hu, X., Zhu, L., Qin, J., Fu, C.W., Heng, P.A.: Recurrently aggregating deep features for salient object detection. In: Thirty-Second AAAI Conference on Artificial Intelligence. (2018)

39. Liu, N., Han, J., Yang, M.H.: Picanet: Learning pixel-wise contextual attention for saliency detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 3089–3098
40. Zhang, L., Dai, J., Lu, H., He, Y., Wang, G.: A bi-directional message passing model for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 1741–1750
41. Wang, T., Borji, A., Zhang, L., Zhang, P., Lu, H.: A stagewise refinement model for detecting salient objects in images. In: Proceedings of the IEEE International Conference on Computer Vision. (2017) 4019–4028
42. Luo, Z., Mishra, A., Achkar, A., Eichel, J., Li, S., Jodoin, P.M.: Non-local deep features for salient object detection. In: Proceedings of the IEEE Conference on computer vision and pattern recognition. (2017) 6609–6617
43. Zhang, P., Wang, D., Lu, H., Wang, H., Ruan, X.: Amulet: Aggregating multi-level convolutional features for salient object detection. In: Proceedings of the IEEE International Conference on Computer Vision. (2017) 202–211
44. Zhang, P., Wang, D., Lu, H., Wang, H., Yin, B.: Learning uncertain convolutional features for accurate saliency detection. In: Proceedings of the IEEE International Conference on computer vision. (2017) 212–221