# Progressively Guided Alternate Refinement Network for RGB-D Salient Object Detection

Shuhan Chen[1][0000−0002−0094−5157] and Yun Fu[2][0000−0002−5098−2853]

[1] School of Information Engineering, Yangzhou University, Yangzhou, China
[2] Department of ECE and Khoury College of Computer Science, Northeastern University, Boston, USA
shchen@yzu.edu.cn, yunfu@ece.neu.edu

**Abstract.** In this paper, we aim to develop an efficient and compact deep network for RGB-D salient object detection, where the depth image provides complementary information to boost performance in complex scenarios. Starting from a coarse initial prediction by a multi-scale residual block, we propose a progressively guided alternate refinement network to refine it. Instead of using ImageNet pre-trained backbone network, we first construct a lightweight depth stream by learning from scratch, which can extract complementary features more efficiently with less redundancy. Then, different from the existing fusion based methods, RGB and depth features are fed into proposed guided residual (GR) blocks alternately to reduce their mutual degradation. By assigning progressive guidance in the stacked GR blocks within each side-output, the false detection and missing parts can be well remedied. Extensive experiments on seven benchmark datasets demonstrate that our model outperforms existing state-of-the-art approaches by a large margin, and also shows superiority in efficiency (**71 FPS**) and model size (**64.9 MB**).

**Keywords:** RGB-D Salient Object Detection · Lightweight Depth Stream · Alternate Refinement · Progressive Guidance

## 1 Introduction

The goal of salient object detection (SOD) is to detect and segment the objects or regions in an image or video [15] that visually attract human attention most. It is usually serves as a pre-processing step to benefit a lot of vision tasks, such as image-sentence matching [23], weakly-supervised semantic segmentation [51], few-shot learning [54], to name a few. Benefiting from the rapid development of deep convolutional neural networks (CNNs), it has achieved profound progresses recently. Nevertheless, it is still very challenging in some complex scenes, such as low contrast, objects sharing similar appearance with its surroundings [11].

RGB-D cameras are now easily available with low-price and high-performance, such as RealSense, Kinect, which can provide depth image that contains necessary geometric information. Utilizing depth additional to the RGB image could potentially improve the performance in the above challenging cases, which is

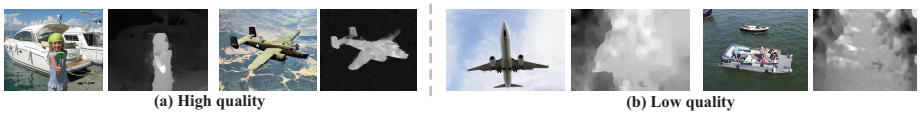



**Fig. 1.** Example RGB images with corresponding depth images. (a) High quality depth images successfully pop out salient objects, thus can be seen as mid-level or high-level feature maps. (b) Low quality depth images are cluttered thus may be harmful for the prediction.

also proven to be an effective way in the applications of object detection [27], semantic segmentation [32], and crowd counting [31].

Although several novel CNN-based SOD approaches [42][62] have been proposed for RGB-D data recently, the optimal way to fuse RGB and depth information remains an open issue, which lies in two aspects: model incompatibility and redundancy, low-quality depth map. Most of the existing fusion strategies can be classified into early fusion [45][47], late fusion [20][14], and middle fusion [1][3][2][42]. Recent researches mainly focus on the middle fusion where a separate backbone network pre-trained from ImageNet [9] is usually utilized to extract depth features, which may causes incompatible problem due to the inherent modality difference between RGB and depth image [62]. Besides that, such two-stream framework doubles the number of model parameters and computation cost, which is not efficient and also contains much redundancy. Furthermore, depth maps may vary in qualities due to the limitations of the depth sensors, high-quality depth map can well pop-out salient objects with well-defined closed boundaries, while low-quality ones are cluttered and may be noisy to the prediction, as shown in Fig. 1.

To address the above issues, we first construct a depth stream to extract depth features by learning from scratch without using pre-trained backbone network. As we know, RGB and depth have very different properties. Depth only captures the spatial structure and 3D layout of objects, without any texture details, thus contains much less information than RGB image. Since lacking low-level information, it is redundant to use pre-trained backbone network to extract features, especially these shallow layers. As seen in Fig. 1, the high quality depth image can be seen as a mid-level or high-level feature map. Based on this observation, we design a lightweight depth stream to capture complementary high-level features only, specifically, four convolutional layers are sufficient to achieve it. Therefore, it is not only much more compact and efficient than existing two-backbone based models, but also without the incompatible problem.

Instead of directly fusing RGB features and depth features (*e.g.*, concatenation or summation), which may degrade the confident RGB features especially when the input depth is noisy, we propose a alternate refinement strategy to incorporate them separately. The whole architecture of the proposed network follows a coarse-to-fine framework which starts from a initial prediction generated by a proposed Multi-Scale Residual (MSR) block. Then, it is refined progressively from deep side-outputs to shallow ones. To alleviate the informa-

tion dilution in the refinement process, we propose a novel guided residual (GR) block where input prediction map is used to guide input feature to generate refined prediction and refined feature, which will be further fed into a following GR block for subsequent refinement. By assigning different guidance roles into different stacked GR blocks within each side-output, the input coarse prediction can be refined progressively into more complete and accurate.

Experimental results over 7 benchmark datasets demonstrate that our model significantly outperforms state-of-the-art approaches, and also with advantages in efficiency (**71 FPS**) and compactness (**64.9 MB**). In summary, our main contributions can be concluded as follows:

- We construct a lightweight depth stream to extract complementary depth features, which is much more compact and efficient than using pre-trained backbone network while without the incompatible problem.
- We propose an alternate refinement strategy by feeding RGB feature and depth feature alternately into GR blocks, in this way to avoid breaking the good property of the confident RGB feature when the depth is low quality.
- We further design a guided residual block to address the information dilution issue, where an input prediction is used as a guidance to generate both refined feature and refined prediction. By stacking them with progressive guidance, the missing parts and false predictions can be well remedied.

## 2 Related Work

### 2.1 RGB Salient Object Detection

**Coarse-to-Fine.** Before deep learning, salient regions are usually discovered from less ambiguous regions to difficult regions [19][57][7]. Following this idea, coarse-to-fine frameworks are widely explored in recent CNNs based works. Liu [35] first proposed a hierarchical recurrent convolutional neural network for refinement in a global to local and coarse to fine pipeline. Using a eye fixation map as initial prediction, Wang *et al.* [50] proposed a hierarchy of convolutional LSTMs to progressively optimize it in a top-down manner. Chen *et al.* [4][5] applied side-output residual learning for refinement, which is guided by a novel reverse attention block. While in [17], attentive feedback module was designed for better guidance. Besides the above top-down guidance, Wang *et al.* [49] further integrated bottom-up inference in an iterative and cooperative manner for recurrent refinement. In this paper, we also follow the coarse-to-fine pipeline, the difference is that we learn multiple residuals in each side-output with progressive guidance, which can better remedy the missing object parts and false detection.

**Top-down Guidance.** The deep layer contains high-level semantic information, which can be used as a guidance to help shallow layers filter out noisy distraction [6]. Such a top-down guidance manner was also widely applied in existing methods. Deep prediction maps are concatenated with shallow prediction by short connections for guidance in [22], and used to erase its corresponding regions in shallow feature to guide residual learning in [4]. While in [52], it was

utilized to weight shallow feature by the proposed holistic attention module. In [36], it was applied into each group of the divided shallow convolutional feature to further promote its guidance role. Liu *et al.* [34] built a global guidance module to transmit the location information into different shallow layers. Zhang *et al.* [58] made a further step by leveraging captioning information to boost SOD. There are also several recent works following this effective strategy, such as [63][39][56]. While in our model, the deep prediction is used as guidance both for feature refinement and prediction refinement. Furthermore, a progressive guidance strategy is proposed to address the feature dilution issue.

### 2.2 RGB-D Salient Object Detection

**Hand-crafted based.** Early works are all focusing on various hand-crafted features, including multi-contextual contrast [41], anisotropic center-surround difference [24], local background enclosure [16], and so on.

**CNNs based.** To increase feature representation ability, CNNs is widely applied and has dominated this area in recent years. As mentioned above, these approaches can be roughly divided into early fusion, middle fusion, and late fusion. Early fusion regards the depth map as a additional channel to concatenate with RGB as initial input, *e.g.*, [47]. Late fusion applies two separate backbone network for RGB and depth to generate individual features or predictions which are fused together for final prediction, such as [20][14]. Most of recent works focused on the middle fusion scheme, which incorporate multi-scale RGB features and depth features in different manners. A complementarity-aware fusion module was proposed in [1]. Piao *et al.* [42] designed a depth refinement block to fuse multi-level paired depth and RGB cues. Instead of cross-modal feature fusion, Zhao *et al.* [62] addressed it in a different manner by integrating enhanced depth cues to weight RGB features for better representation. In [43], an asymmetrical two-stream architecture based on knowledge distillation was proposed for light field SOD. Different from them, we construct a lightweight depth stream by learning from scratch, whose depth features are fed for refinement separately.

## 3 The Proposed Network

We first present the overall architecture of the proposed alternate refinement network, and then introduce its main components in detail, including MSR block to generate coarse initial prediction, and GR block with progressive guidance. Finally, we also discuss the differences with related networks.

### 3.1 The Overall Architecture

Our network follows the existing coarse-to-fine refinement framework as seen in Fig. 2. Given a coarse initial prediction generated by our MSR block, we apply our GR block to refine it progressively by combing multi-level convolutional features from RGB and depth streams alternately. Considering the modal gap

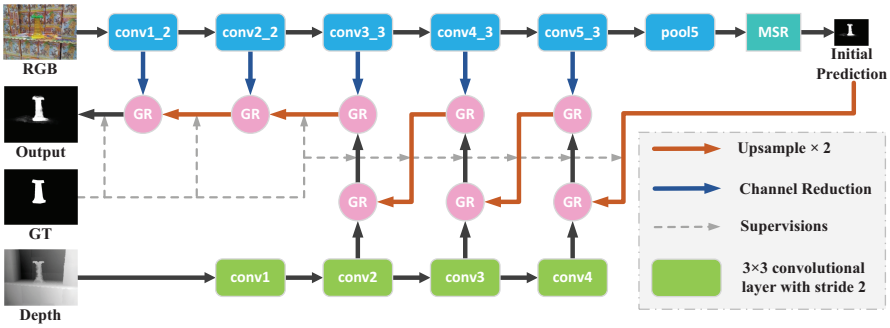



**Fig. 2.** The overall architecture of the proposed network, where RGB feature and depth feature are fed into GR blocks **alternately** for refinement. Here, we only show single GR block in each side-output for clarity. Detailed structures of MSR and GR are illustrated in Fig. 3 and Fig. 4 respectively.

between RGB and depth, furthermore, the quality of the depth varies tremendously across different scenarios due to the limitation of depth sensors, we don't directly fuse the RGB and depth features, instead, they are fed into our network alternately to reduce their mutual degradation for better refinement. Finally, we apply deep supervisions on each side-output and train the whole network in an end-to-end manner, only with standard binary cross entropy loss.

**RGB stream.** We utilize VGG16 [46] as backbone network to extract multi-level RGB features, where {`conv1_2`, conv2_2, `conv3_3`, `conv4_3`, `conv5_3`} are chosen as side-output features, which have {1, 1/2, 1/4, 1/8, 1/16} of the input image resolution respectively. We first apply $1 \times 1$ convolutional layers to reduce their dimensions into {16, 32, 64, 64, 64} for efficiency. Then, these side-output features (denoted as $F_1$, $F_2$, $F_3$, $F_5$, $F_7$) are used for subsequent refinement.

**Depth stream.** Instead of using pre-trained backbone network as most of the existing works did, we construct a light-weight depth stream to extract complementary features, which only consists of four cascaded $3 \times 3$ convolutional layers with 64 channels and stride 2. The last three layers are selected as high-level side-output features for refinement, which are denoted as $F_4$, $F_6$, $F_8$, with {1/4, 1/8, 1/16} of the input image resolution respectively.

### 3.2 Multi-Scale Residual Block

Since the scale of salient objects vary from large to small, which implies that the model needs to capture information at different contexts in order to detect objects reliably. Although the backbone network has a large enough theoretical receptive field to cover most of the large objects, the effective receptive field is smaller than the theoretical receptive field as demonstrated in [37]. Inspired from [30], we design a multi-scale residual block to address the scale issue for SOD. The proposed MSR block is embedded after "`pool5`" and consists of three parallel branches in which each shares the same residual structure except the dilation rate. Each residual block consists of three convolutions with kernel size

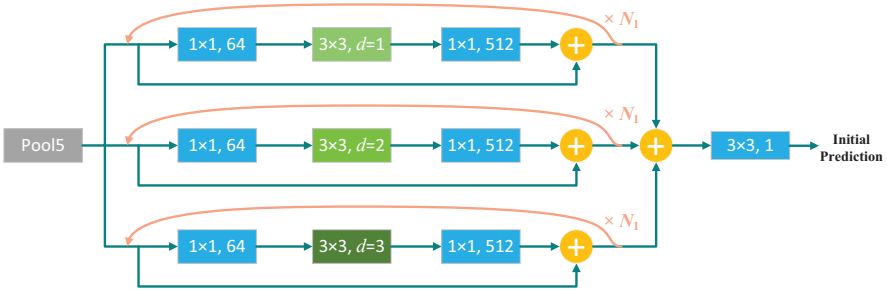



**Fig. 3.** The proposed multi-scale residual block. "$d$" denotes dilation rate.

$1 \times 1$, $3 \times 3$, and $1 \times 1$. The dilation rates for the $3 \times 3$ convolutional layers are 1, 2, and 3 respectively, as shown in Fig. 3. Instead of stacking such residual block to increase receptive field, we implement it in a recurrent manner to reduce the number of parameters, which has been widely applied in previous works [60][10][29]. The total recurrent iteration is $N_1$ for each branch. Finally, all the branches are added together then fed into a $3 \times 3$ convolutional layer to produce a single channel initial prediction.

Although shares similar structure, the proposed MSR differs from [30] in the following two aspects. Firstly, since the scale-aware ground truth is not easy to obtain in SOD, we don't share weights among different branches but different stacked blocks in each branch. Secondly, these branches are fused together for the initial prediction.

### 3.3 Guided Residual Block

As we know, different layers of deep CNNs learn different scale features, shallow layers capture low-level structure cues while deep layers capture high-level semantic information. Based on this observation, various fusion strategies were proposed to combine their complementary cues, such as short connection [22], skip connection [33], residual connection [21][10][61]. However, the high-level information of deep layer may be gradually diluted in the fusion process, especially when combing with the noisy shallow features. To address it, we design a novel guided residual block which composes of two parts: split-and-concatenate (SC) operation, and dual residual learning. As illustrated in Fig. 4, given an input feature and prediction map, it outputs refined alternatives.

**Split-and-Concatenate.** Given the convolutional feature $F$ with $C$ channels and prediction map $S$ as inputs, we first split $F$ into $g$ groups, each of which has $c$ channels. Then $S$ is utilized as a guidance feature map to be concatenated with each split feature maps. After concatenation, we obtain a $C + g$ channel feature. Based on it, the SC operation can be formulated as:

$$F^1, ..., F^j, ..., F^g = \text{Split}(F), j \in 1, 2, ..., g, \tag{1}$$

$$F_{cat} = \text{Cat}(F^1, S, ..., F^j, S, ..., F^g, S), \tag{2}$$

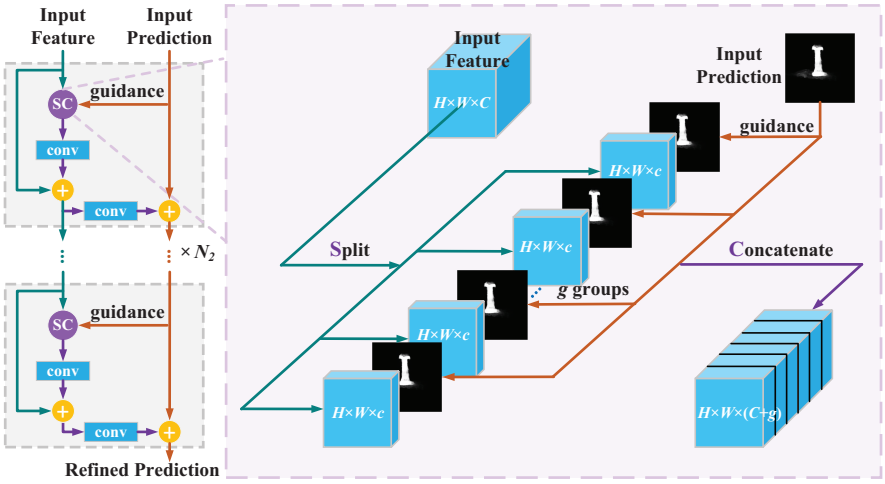



**Fig. 4.** The proposed guided residual blocks (dashed bounding boxes on the left), which are stacked with progressive guidance. SC denotes split-and-concatenate operation (dashed bounding box on the right).

where Cat denotes concatenate operation and $F_{cat}$ is the concatenated feature.

**Dual Residual Learning.** After SC operation, we feed the concatenated feature $F_{cat}$ into a $3 \times 3$ convolutional layer for guided learning and reducing channel number into $C$, then are added with the input feature as refined output feature. Thus, the first residual learning can be formulated as:

$$\hat{F} = F + \text{Conv}(F_{cat}; \theta_1), \tag{3}$$

where $\text{Conv}(*; \theta)$ is the convolution operation with parameter $\theta$. Another $3 \times 3$ convolutional layer is further applied to produce a single channel residual prediction. Based on it, we can obtain the refined prediction map $\hat{S}$ by:

$$\hat{S} = S + \text{Conv}(\hat{F}; \theta_2). \tag{4}$$

$\hat{F}$ and $\hat{S}$ will be further fed into subsequent GR block for guided residual learning.

### 3.4 Progressive Guidance

Based on the above GR block, we propose a progressive residual refinement framework. Specifically, in each side-output, we stack $N_2$ GR blocks, which is set to 3 in this paper, and the first GR block takes $F_i^0$ and $S_i^0$ as input feature and prediction map:

$$\begin{cases} S_i^0 = \text{Up}(S_{i+1}, F_i), \text{ if } i = 1, 2, 4, 6, 8; & (5) \\ S_i^0 = S_{i+1}, \text{ if } i = 3, 5, 7; & (6) \end{cases}$$

**Table 1.** The detailed setting of different guidance styles, including uniform guidance and progressive guidance. {*,*,*} represents the channel number $c$ in each split group from side-output 1 to side-output 3. The rest are the same with side-output 3. $\text{GR}^r$ denotes the $r$th GR block in each side-output.

| Guidance Style | No. | $\text{GR}^1$ | $\text{GR}^2$ | $\text{GR}^3$ |
|---|---|---|---|---|
| Uniform Guidance | 1 | | $c$={16,32,64} | |
| | 2 | | $c$={8,8,8} | |
| | 3 | | $c$={4,4,4} | |
| | 4 | | $c$={1,1,1} | |
| Progressive Guidance | 5 | $c$={16,32,64} | $c$={8,8,8} | $c$={4,4,4} |
| | 6 | $c$={16,32,64} | $c$={8,8,8} | $c$={1,1,1} |
| | 7 | $c$={16,32,64} | $c$={4,4,4} | $c$={1,1,1} |
| | 8 | $c$={8,8,8} | $c$={4,4,4} | $c$={1,1,1} |

in which $\text{Up}(x, y)$ represents bilinear interpolation operation that upsamples $x$ to the same size as $y$, and $i$ denotes side-output stage. Then the output of the first GR block can be denoted as $\hat{F}_i^1$ and $\hat{S}_i^1$, which will be fed into the following GR block. The last GR block only outputs the refined prediction $S_i = \hat{S}_i^{N_2}$ as shown in Fig. 4. $S_1$ is fed into a sigmoid layer as final output.

The channel number $c$ in each split group or group number $g$ is essential for guidance. We can define different guidance styles by varying $c$ or $g$. If $g = 1$, as [10] did, the guidance role is very weak due to the imbalanced channels ($C$ versus 1). In [36], $c$ is set to 4 in all the side-outputs, which can be seen as a medium guidance. Extremely, the guidance role is very strong when we set $c$ to 1. Different from them, we first define uniform guidance by sharing the same guidance role in all the stacked blocks, as listed in Table 1. Since the prediction map will becomes more and more accurate in the refinement process, we further define progressive guidance by gradually increasing the guidance role. We will conduct ablation experiments to investigate the best setting in Section 4.2.

Such a progressive residual refinement inherits the following good properties. The stacked residual units establish multiple shortcut connections between each side-output prediction and the ground truth, which enables it easier to remedy the missing object parts and detection errors. Extremely, with the strong supervision on each side-output, the error is approximately equal to zero if there is no useful information in the input feature, *e.g.*, when the depth image is low quality. In this way, we can greatly reduce its noisy distraction, thus leads to more accurate detection. Furthermore, such residual units also enhance the input feature gradually for better refinement.

### 3.5 Difference to Other Networks

Although shares the same split-and-concatenate operation, the proposed GR block differs from the group guidance module (GGM) [36] in two aspects. (1) GGM apply group convolution on the output of SC, which only focuses on each

split group for guidance. Different from it, the convolution is performed on all the concatenated feature maps in our GR, which benefits the information passing among different groups. (2) The channel number in each split group is fixed to 4 in GGM. While our GR blocks are stacked with progressive guidance by varying different $c$. Our network also differs other residual learning based architectures, *e.g.*, RAS [4][5], $R^3$Net [10]. Firstly, we learn dual residuals progressively in each side-output which can better remedy the missing object parts and false detection in the initial prediction. Secondly, the prediction maps are progressively applied for guidance during residual refinement. The effectiveness and superiority will be verified in the following section.

## 4 Experimental Results

### 4.1 Experimental Setup

**Datasets.** We adopt 7 widely used RGB-D benchmark datasets for evaluation, including NJUD [24], NLPR [41], DES [8], STERE [38], LFSD [28], DUT [42], and SIP [14], which contain 1985, 1000, 135, 1000, 100, 1200, 929 well annotated images, respectively. Among them, SIP is a recent collected human activities oriented dataset with high image resolution (744×992). To make a fair comparison, we follow the same training settings as existing works [20][1][2][14], which consists of 1485 samples from NJUD and 700 samples from NLPR. To reduce over-fitting risk, we augment the input image by random horizontal flipping and rotating (0°, 90°, 180°, 270°), which increases the training images by four times.

**Evaluation Metrics.** We adopt five widely applied metrics for comprehensive evaluation, *i.e.*, precision-recall (PR) curve, F-measure ($F_\beta$), S-measure [12] ($S_\alpha$), E-measure [13] ($E_\xi$), and mean absolute error ($M$). Specifically, PR curve is plotted via pairs of precision and recall values which are calculated by comparing the binary saliency map with its ground truth. $F_\beta$ is an overall metric and only its maximum value is reported here, where $\beta^2$ is set to 0.3 to emphasize the precision over recall. $S_\alpha$ and $E_\xi$ are two recent proposed metrics which evaluate the spatial structure similarities, local pixel matching and image-level statistics information, respectively. Higher scores of $E_\xi$, $S_\alpha$, and $F_\beta$ indicate better performance, while lower for $M$,.

**Implementation Details.** We implemented our method in PyTorch [40] and on a PC with single NVIDIA TITAN Xp GPU. All the images are resized to 352×352 both for training and inferring. The depth image needs to be normalized into [0, 1]. The proposed model is trained by Adam optimizer [25] with the following hyper-parameters: batch size (10), epochs (30), initial learning rate (1e-4), which is decreased by 10 after 25 epochs. The parameters of the backbone network in the RGB stream is initialized by VGG16 [46], while the others are using the default setting of the Pytorch. We will release the source code for research purpose on http://shuhanchen.net.

**Table 2.** Quantitative comparison of different settings in Table 1.

| No. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| $E_\xi \uparrow$ | 0.903 | 0.903 | 0.903 | 0.904 | 0.903 | **0.908** | 0.907 | 0.905 |
| $S_\alpha \uparrow$ | 0.866 | 0.869 | 0.869 | 0.870 | 0.867 | **0.875** | 0.871 | 0.871 |
| $F_\beta \uparrow$ | 0.845 | 0.845 | 0.845 | 0.846 | 0.845 | **0.848** | 0.847 | 0.842 |
| $M \downarrow$ | 0.062 | 0.062 | 0.062 | 0.060 | 0.061 | 0.059 | **0.058** | 0.059 |

**Table 3.** Quantitative comparison with different ablation settings. R and D denote RGB stream and depth stream respectively. St: stacking 7 MSR blocks; Re: proposed recurrent strategy. MS: model size (MB).

| | St | Re | Cat | AR | R | R+D | VGG16 | Ours |
|---|---|---|---|---|---|---|---|---|
| $E_\xi \uparrow$ | **0.907** | **0.907** | 0.899 | **0.908** | 0.886 | **0.908** | 0.896 | **0.908** |
| $S_\alpha \uparrow$ | 0.871 | **0.872** | 0.863 | **0.875** | 0.846 | **0.875** | 0.857 | **0.875** |
| $F_\beta \uparrow$ | 0.850 | **0.851** | 0.840 | **0.848** | 0.814 | **0.848** | 0.837 | **0.848** |
| $M \downarrow$ | **0.059** | **0.059** | 0.064 | **0.059** | 0.072 | **0.059** | 0.065 | **0.059** |
| MS $\downarrow$ | 72.3 | **64.9** | **63.0** | 64.9 | **62.5** | 64.9 | 123.6 | **64.9** |

### 4.2 Ablation Analyses

We first investigate different design options and the effectiveness of different components in the proposed network on a recent challenging dataset SIP [14].

**Recurrent strategy.** To verify the effectiveness of the recurrent strategy in MSR, we first made an experiment by comparing with stacking 7 blocks with 7 iterations. Here we adopt the No. 1 setting in Table 1 as guidance style. As can be seen in Table 3, their performance are almost the same, but the recurrent strategy achieved more compact model size. We further conduct ablation study to explore how many recurrent iterations are needed in MSR by varying it $N_1$ from 1 to 7. The results in Fig. 5(f) show that when $N_1$ grows beyond 5, the performance becomes stable. Therefore, for efficiency, we adopt the recurrent strategy and set $N_1$ to 5 in the following experiments.

**Guidance style.** To investigate the best setting of the guidance style, we compare the performance of all the listed settings in Table 1. From the results in Table 2, we can observe that progressive guidance shows better performance than uniform guidance, which supports our claim that the guidance role should be strengthened with the progressively refined prediction map. The No.6 setting that achieved best performance was adopt as our final guidance strategy.

**Alternate refinement.** We further conduct experiment to verify the proposed alternate refinement (AR) strategy by comparing with directly concatenating (Cat) the RGB and depth features. As shown in Table 3, our proposed AR strategy performs better than Cat, which demonstrates our analysis that Cat may break the good property of the RGB features.

**Depth stream.** We first evaluate the effectiveness of the constructed depth stream by removing it for comparison. As seen in Table 3, the performance

**Table 4.** Quantitative comparison with different side-output depth features.

| conv4 | conv3 | conv2 | conv1 | $E_\xi \uparrow$ | $S_\alpha \uparrow$ | $F_\beta \uparrow$ | $M \downarrow$ |
|---|---|---|---|---|---|---|---|
| ✓ | | | | 0.892 | 0.855 | 0.828 | 0.067 |
| ✓ | ✓ | | | 0.897 | 0.860 | 0.834 | 0.066 |
| ✓ | ✓ | ✓ | | **0.908** | **0.875** | **0.848** | **0.059** |
| ✓ | ✓ | ✓ | ✓ | 0.907 | 0.873 | **0.848** | **0.059** |

can be greatly improved when combining the depth stream, which indicates the good ability to capture complementary information by our constructed depth stream. It is also worth to note that the performance is still comparable with state-of-the-art model when only using RGB stream, which further confirms the effectiveness of our progressive residual refinement framework.

Since most of the previous works using two pre-trained backbone networks to extract RGB and depth features respectively, we also made another experiment by replacing the proposed depth stream with VGG16 [46]. As a result, the model size and training time (4 hours) are dramatically increased with the decrease of the quantitative performance as shown in Table 3. Therefore, our proposed depth stream is a better choice in extracting complementary depth features.

Finally, to investigate how many depth features are sufficient for the proposed network, we separately evaluate the performance by combing different side-output depth features. We can clearly observe from Table 4 that the performance is gradually improved with the incorporation of more side-output depth features until "`conv2`". Further incorporating "`conv1`" doesn't bring performance gain, which supports our claim that depth image can be seen as a mid-level or high-level feature map, therefore, there is no need to explore low-level features from it with additional convolutional layers. Therefore, three side-output depth features are sufficient to capture the complementary cues.

### 4.3 Comparison with State-of-the-arts

We compare our model with 10 state-of-the-arts, consisting of 3 traditional methods: LHM [41], ACSD [24], LBE [16]; and 7 CNNs-based methods: DF [45], CTMF [20], MMCI [3], TAN [2], PCAN [1], CPFP [62], DMRA [42]. Note that all the results of the compared approaches are reproduced by running source codes or pre-computed by the authors. In addition, we also trained a model (marked with $*$) using the same trainset with DMRA [42] for fair comparison.

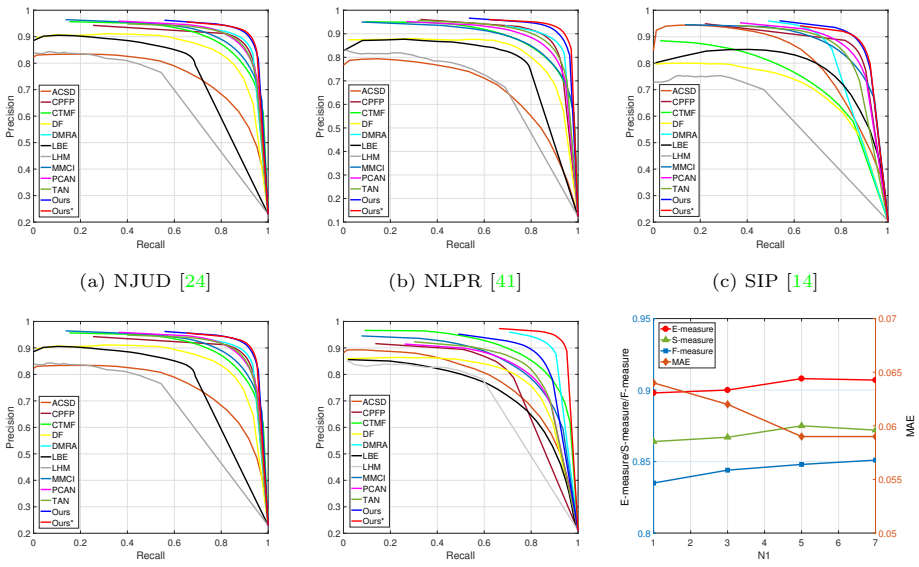
**Quantitative Evaluation.** The quantitative comparison results in terms of 4 evaluation metrics on 7 datasets are reported in Table 5. As can be clearly observed that the proposed network significantly outperforms the competing methods across all the datasets in all the metrics except $E_\xi$. Comparing with the recent state-of-the-art model DMRA [42], our approach increases its $S_\alpha$ and $F_\beta$ scores by an average of **2.8%** and **2.1%**, decreases the $M$ by an average of **1.0%**, which clearly indicates the good consistence with the ground truth. We

**Table 5.** Quantitative comparison including $E_\xi$, $S_\alpha$, $F_\beta$, and $M$, over seven widely evaluated datasets. ↑ & ↓ represent higher and lower is better, respectively. $*$ denotes the models are trained on NJUD [24]+NLPR [41]+DUT [42], the rest are trained on NJUD [24]+NLPR [41]. The best three scores are highlighted in **red**, **blue**, and **green** respectively.

| | Metric | LHM [41] | ACSD [24] | LBE [16] | DF [45] | CTMF [20] | MMCI [3] | TAN [2] | PCAN [1] | CPFP [62] | Ours | DMRA [42]* | Ours * |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NJUD [24] | $E_\xi$ ↑ | .711 | .790 | .796 | .839 | .864 | .878 | .893 | .896 | .895 | **.914** | **.908** | **.916** |
| | $S_\alpha$ ↑ | .522 | .703 | .700 | .768 | .849 | .859 | .878 | .877 | .878 | **.906** | **.886** | **.909** |
| | $F_\beta$ ↑ | .636 | .695 | .734 | .783 | .788 | .813 | .844 | .844 | .837 | **.883** | **.872** | **.893** |
| | $M$ ↓ | .199 | .198 | .149 | .136 | .085 | .079 | .060 | .059 | .053 | **.045** | **.051** | **.042** |
| NLPR [41] | $E_\xi$ ↑ | .819 | .752 | .868 | .884 | .869 | .872 | .916 | .916 | **.924** | **.948** | **.941** | **.955** |
| | $S_\alpha$ ↑ | .631 | .684 | .777 | .806 | .860 | .856 | .886 | .874 | .888 | **.918** | **.899** | **.930** |
| | $F_\beta$ ↑ | .665 | .548 | .747 | .759 | .723 | .730 | .796 | .795 | .822 | **.871** | **.854** | **.885** |
| | $M$ ↓ | .103 | .171 | .073 | .079 | .056 | .059 | .041 | .044 | .036 | **.028** | **.031** | **.024** |
| DES [8] | $E_\xi$ ↑ | .761 | .855 | .911 | .877 | .911 | .904 | .919 | .912 | .927 | **.935** | **.944** | **.939** |
| | $S_\alpha$ ↑ | .578 | .728 | .703 | .752 | .863 | .848 | .858 | .842 | .872 | **.894** | **.900** | **.913** |
| | $F_\beta$ ↑ | .631 | .717 | .796 | .753 | .778 | .762 | .795 | .782 | .829 | **.870** | **.866** | **.880** |
| | $M$ ↓ | .114 | .169 | .208 | .093 | .055 | .065 | .046 | .049 | .038 | **.032** | **.030** | **.026** |
| STERE [38] | $E_\xi$ ↑ | .770 | .793 | .749 | .838 | .864 | .901 | .906 | .897 | .903 | **.917** | **.920** | **.919** |
| | $S_\alpha$ ↑ | .562 | .692 | .660 | .757 | .848 | .873 | .871 | .875 | .879 | **.903** | **.886** | **.907** |
| | $F_\beta$ ↑ | .703 | .661 | .595 | .742 | .771 | .829 | .835 | .826 | .830 | **.872** | **.867** | **.880** |
| | $M$ ↓ | .172 | .200 | .250 | .141 | .086 | .068 | .060 | .064 | .051 | **.044** | **.047** | **.041** |
| SIP [14] | $E_\xi$ ↑ | .719 | .827 | .841 | .794 | .824 | .886 | **.893** | **.899** | **.899** | **.908** | .863 | **.908** |
| | $S_\alpha$ ↑ | .511 | .732 | .727 | .653 | .716 | .833 | .835 | .842 | **.850** | **.875** | .806 | **.876** |
| | $F_\beta$ ↑ | .592 | .727 | .733 | .673 | .684 | .795 | .809 | **.825** | .819 | **.848** | .819 | **.854** |
| | $M$ ↓ | .184 | .172 | .200 | .185 | .139 | .086 | .075 | .071 | **.064** | **.059** | .085 | **.055** |
| DUT [42] | $E_\xi$ ↑ | .756 | .814 | .785 | .848 | .884 | .855 | .866 | .858 | .815 | **.888** | **.927** | **.944** |
| | $S_\alpha$ ↑ | .551 | .706 | .679 | .733 | .834 | .791 | .808 | .801 | .749 | **.849** | **.889** | **.920** |
| | $F_\beta$ ↑ | .683 | .699 | .668 | .764 | .792 | .753 | .779 | .760 | .736 | **.829** | **.884** | **.914** |
| | $M$ ↓ | .179 | .181 | .236 | .144 | .097 | .113 | .093 | .100 | .100 | **.069** | **.048** | **.035** |
| LFSD [28] | $E_\xi$ ↑ | .736 | .801 | .770 | .844 | .851 | .840 | .845 | .842 | .867 | **.869** | **.899** | **.889** |
| | $S_\alpha$ ↑ | .557 | .734 | .736 | .791 | .796 | .787 | .801 | .794 | .828 | **.833** | **.847** | **.853** |
| | $F_\beta$ ↑ | .718 | .755 | .708 | .806 | .782 | .779 | .794 | .792 | .813 | **.830** | **.849** | **.852** |
| | $M$ ↓ | .211 | .188 | .208 | .138 | .119 | .132 | .111 | .112 | **.088** | .093 | **.075** | **.074** |

also perform much better when comparing with CPFP [62] which doesn't use pre-trained backbone network to extract depth features too. We analyze that their proposed contrast prior may also break the good property of the RGB features when the depth image is low quality, while such issue can be well alleviated by our method. It is also worth to note that our model trained on NJUD+NLPR still performs better than DMRA on some datasets, which further demonstrates the superiority and effectiveness of the proposed approach. We also plot the PR curves for comparison on five large datasets. As illustrated in Fig. 5, we consistently achieve the best performance especially at a high level of recall.

**Qualitative Evaluation.** We further illustrate visual examples of several representative images in different challenging scenarios to show the advantage of our method, *i.e.*, low contrast in RGB or depth image ($1^{st}$-$2^{nd}$ rows), low quality depth map ($3^{rd}$-$4^{th}$ rows), complex scene ($5^{th}$ row), multiple (small) objects ($6^{th}$-$7^{th}$ rows), and large object ($8^{th}$ row). As can be seen clearly in Fig.6 that all these cases are very challenging to the existing methods. Nevertheless, thanks to the proposed alternate refinement strategy, our model can well capture the complementary cues from the depth image, therefore, we can successfully




(a) NJUD [24] (b) NLPR [41] (c) SIP [14]

(d) STERE [38] (e) DUT [42] (f) SIP [14]

**Fig. 5.** (a)-(e): Precision-recall curves comparison. (f) Quantitative comparison of different recurrent iterations in the proposed MSR block.

**Table 6.** Running speed and model size comparisons with recent models.

| Method | Platform | Image Size | FPS ↑ | MS (MB) ↓ |
|---|---|---|---|---|
| CPFP [62] | Caffe | 400×300 | 10 | 291.9 |
| DMRA [42] | Pytorch | 256×256 | 16 | 238.8 |
| Ours | Pytorch | 352×352 | **71** | **64.9** |

highlight salient objects in these images, and also will not be distracted by the low quality depth maps. Furthermore, contributed by the proposed progressive guidance, the missing object parts and false detection can be well remedied, thus leads to more complete and accurate detection.

**Timing and Model Size.** The proposed network is also very efficient and compact. When trained on the NJUD+NLPR datasets with 2185×4 images, our network only takes about 2.5 hours to train for 30 epochs. During the inference stage, contributed by the constructed lightweight depth stream, we can run at **71 FPS** only with **64.9 MB** model size, which is much faster and compact than the existing models as shown in Table 6.

## 5 Conclusions

In this paper, we developed a progressively guided alternate refinement network for efficient RGB-D SOD. A lightweight depth stream was first constructed to

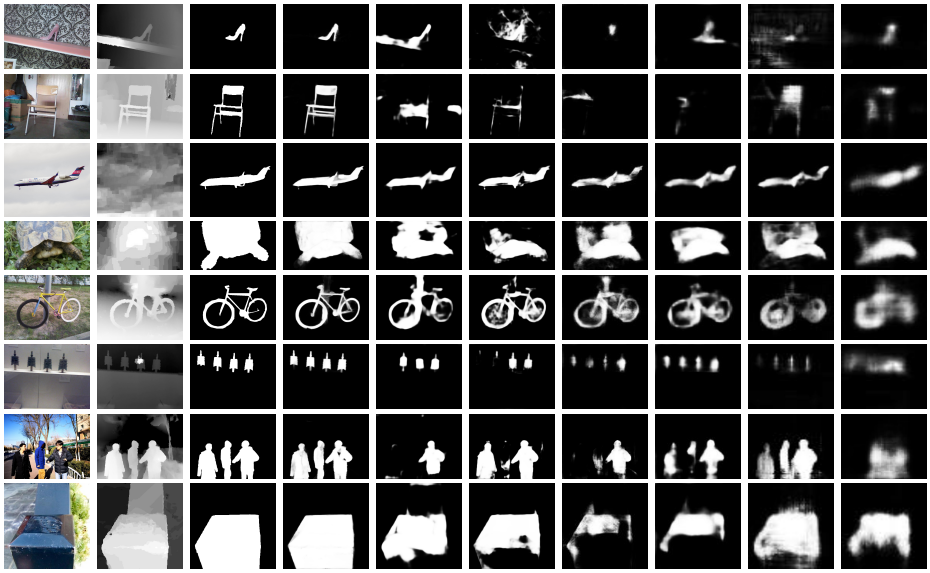



**Fig. 6.** Visual comparisons with state-of-the-art approaches in different challenging scenarios: low contrast in RGB or depth, complex scene, low quality depth map, multiple (small) objects, and large object. As can be seen, all the salient objects can be completely highlighted while with less false detection.

extract complementary depth features by learning from scratch. Starting from a coarse initial prediction by the proposed MSR block, the RGB features and depth features are alternately fed into the designed GR blocks for progressive refinement. Contributed by the alternate refinement strategy, the mutual degradation between RGB and depth features can be well alleviated especially when the depth is low quality. With the help of the proposed progressive guidance, the missing object parts and false detection can be well refined, which resulted in more complete and accurate detection. State-of-the-art performance on 7 benchmark datasets demonstrates their effectiveness, and also shows the superiority in efficiency and compactness. In addition, the proposed network can be flexibly applied for other cross-modal SOD tasks, *e.g.*, RGB-T [48]. Nevertheless, the boundary details are still not accurate enough especially in high resolution images [53], which will be further improved in future works. We also found that some new approaches with high performance are published after this submission, such as ICNet [26], UCNet [55], JL-DCF [18], A2dele [44], and SSF [59]. We will make a more comprehensive comparison in our extended work.

**Acknowledgments.** This research was supported by the National Nature Science Foundation of China (No. 61802336) and China Scholarship Council (CSC) Program. This work was mainly done when Shuhan Chen was visiting Northeastern University as a visiting scholar.

## References


1. Chen, H., Li, Y.: Progressively complementarity-aware fusion network for rgb-d salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3051–3060 (2018) 2, 4, 9, 11, 12
2. Chen, H., Li, Y.: Three-stream attention-aware network for rgb-d salient object detection. IEEE Transactions on Image Processing **28**(6), 2825–2835 (2019) 2, 9, 11, 12
3. 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**, 376–385 (2019) 2, 11, 12
4. Chen, S., Tan, X., Wang, B., Hu, X.: Reverse attention for salient object detection. In: Proceedings of the European Conference on Computer Vision. pp. 234–250 (2018) 3, 9
5. Chen, S., Tan, X., Wang, B., Lu, H., Hu, X., Fu, Y.: Reverse attention-based residual network for salient object detection. IEEE Transactions on Image Processing **29**, 3763–3776 (2020) 3, 9
6. Chen, S., Wang, B., Tan, X., Hu, X.: Embedding attention and residual network for accurate salient object detection. IEEE Transactions on Cybernetics **50**(5), 2050–2062 (2020) 3
7. Chen, S., Zheng, L., Hu, X., Zhou, P.: Discriminative saliency propagation with sink points. Pattern Recognition **60**, 2–12 (2016) 3
8. Cheng, Y., Fu, H., Wei, X., Xiao, J., Cao, X.: Depth enhanced saliency detection method. In: Proceedings of International Conference on Internet Multimedia Computing and Service. pp. 23–27 (2014) 9, 12
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009) 2
10. Deng, Z., Hu, X., Zhu, L., Xu, X., Qin, J., Han, G., Heng, P.A.: $R^3$net: Recurrent residual refinement network for saliency detection. In: Proceedings of the International Joint Conference on Artificial Intelligence. pp. 684–690 (2018) 6, 8, 9
11. 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. pp. 186–202 (2018) 1
12. 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. pp. 4548–4557 (2017) 9
13. Fan, D.P., Gong, C., Cao, Y., Ren, B., Cheng, M.M., Borji, A.: Enhanced-alignment measure for binary foreground map evaluation. In: Proceedings of the International Joint Conference on Artificial Intelligence. pp. 698–704 (2018) 9
14. Fan, D.P., Lin, Z., Zhang, Z., Zhu, M., Cheng, M.M.: Rethinking RGB-D salient object detection: Models, datasets, and large-scale benchmarks. IEEE Transactions on Neural Networks and Learning Systems (2020) 2, 4, 9, 10, 12, 13
15. 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. pp. 8554–8564 (2019) 1
16. Feng, D., Barnes, N., You, S., McCarthy, C.: Local background enclosure for rgb-d salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2343–2350 (2016) 4, 11, 12

17. 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. pp. 1623–1632 (2019) 3
18. Fu, K., Fan, D.P., Ji, G.P., Zhao, Q.: Jl-dcf: Joint learning and densely-cooperative fusion framework for rgb-d salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3052–3062 (2020) 14
19. Gong, C., Tao, D., Liu, W., Maybank, S.J., Fang, M., Fu, K., Yang, J.: Saliency propagation from simple to difficult. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2531–2539 (2015) 3
20. Han, J., Chen, H., Liu, N., Yan, C., Li, X.: Cnns-based rgb-d saliency detection via cross-view transfer and multiview fusion. IEEE Transactions on Cybernetics **48**(11), 3171–3183 (2017) 2, 4, 9, 11, 12
21. 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. pp. 770–778 (2016) 6
22. Hou, Q., Cheng, M., Hu, X., Borji, A., Tu, Z., Torr, P.: Deeply supervised salient object detection with short connections. IEEE Transactions on Pattern Analysis and Machine Intelligence **41**(4) (2019) 3, 6
23. Ji, Z., Wang, H., Han, J., Pang, Y.: Saliency-guided attention network for image-sentence matching. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 5754–5763 (2019) 1
24. Ju, R., Ge, L., Geng, W., Ren, T., Wu, G.: Depth saliency based on anisotropic center-surround difference. In: Proceedings of the IEEE International Conference on Image Processing. pp. 1115–1119 (2014) 4, 9, 11, 12, 13
25. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Proceedings of the International Conference on Learning Representations (2015) 9
26. Li, G., Liu, Z., Ling, H.: Icnet: Information conversion network for rgb-d based salient object detection. IEEE Transactions on Image Processing **29**, 4873–4884 (2020) 14
27. Li, G., Gan, Y., Wu, H., Xiao, N., Lin, L.: Cross-modal attentional context learning for rgb-d object detection. IEEE Transactions on Image Processing **28**(4), 1591–1601 (2018) 2
28. Li, N., Ye, J., Ji, Y., Ling, H., Yu, J.: Saliency detection on light field. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2806–2813 (2014) 9, 12
29. Li, X., Wu, J., Lin, Z., Liu, H., Zha, H.: Recurrent squeeze-and-excitation context aggregation net for single image deraining. In: Proceedings of the European Conference on Computer Vision. pp. 254–269 (2018) 6
30. Li, Y., Chen, Y., Wang, N., Zhang, Z.: Scale-aware trident networks for object detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6054–6063 (2019) 5, 6
31. Lian, D., Li, J., Zheng, J., Luo, W., Gao, S.: Density map regression guided detection network for rgb-d crowd counting and localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1821–1830 (2019) 2
32. Lin, D., Zhang, R., Ji, Y., Li, P., Huang, H.: Scn: Switchable context network for semantic segmentation of rgb-d images. IEEE Transactions on Cybernetics **50**(3), 1120–1131 (2018) 2

33. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2117–2125 (2017) 6
34. 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. pp. 3917–3926 (2019) 4
35. Liu, N., Han, J.: Dhsnet: Deep hierarchical saliency network for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 678–686 (2016) 3
36. Liu, Y., Han, J., Zhang, Q., Shan, C.: Deep salient object detection with contextual information guidance. IEEE Transactions on Image Processing **29**, 360–374 (2019) 4, 8
37. Luo, W., Li, Y., Urtasun, R., Zemel, R.: Understanding the effective receptive field in deep convolutional neural networks. In: Proceedings of Advances in Neural Information Processing Systems. pp. 4898–4906 (2016) 5
38. Niu, Y., Geng, Y., Li, X., Liu, F.: Leveraging stereopsis for saliency analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 454–461 (2012) 9, 12, 13
39. Pang, Y., Zhao, X., Zhang, L., Lu, H.: Multi-scale interactive network for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9413–9422 (2020) 4
40. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems. pp. 8024–8035 (2019) 9
41. Peng, H., Li, B., Xiong, W., Hu, W., Ji, R.: Rgbd salient object detection: a benchmark and algorithms. In: Proceedings of the European Conference on Computer Vision. pp. 92–109 (2014) 4, 9, 11, 12, 13
42. 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. pp. 7254–7263 (2019) 2, 4, 9, 11, 12, 13
43. Piao, Y., Rong, Z., Zhang, M., Lu, H.: Exploit and replace: An asymmetrical two-stream architecture for versatile light field saliency detection. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 11865–11873 (2020) 4
44. Piao, Y., Rong, Z., Zhang, M., Ren, W., Lu, H.: A2dele: Adaptive and attentive depth distiller for efficient rgb-d salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9060–9069 (2020) 14
45. Qu, L., He, S., Zhang, J., Tian, J., Tang, Y., Yang, Q.: Rgbd salient object detection via deep fusion. IEEE Transactions on Image Processing **26**(5), 2274–2285 (2017) 2, 11, 12
46. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: Proceedings of the International Conference on Learning Representations (2015) 5, 9, 11
47. Song, H., Liu, Z., Du, H., Sun, G., Le Meur, O., Ren, T.: Depth-aware salient object detection and segmentation via multiscale discriminative saliency fusion and bootstrap learning. IEEE Transactions on Image Processing **26**(9), 4204–4216 (2017) 2, 4
48. Tang, J., Fan, D., Wang, X., Tu, Z., Li, C.: Rgbt salient object detection: Benchmark and a novel cooperative ranking approach. IEEE Transactions on Circuits and Systems for Video Technology (2019) 14

49. 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. pp. 5968–5977 (2019) 3
50. Wang, W., Shen, J., Dong, X., Borji, A.: Salient object detection driven by fixation prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1711–1720 (2018) 3
51. Wei, Y., Liang, X., Chen, Y., Shen, X., Cheng, M.M., Feng, J., Zhao, Y., Yan, S.: Stc: A simple to complex framework for weakly-supervised semantic segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(11), 2314–2320 (2016) 1
52. Wu, Z., Su, L., Huang, Q.: Cascaded partial decoder for fast and accurate salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3907–3916 (2019) 3
53. Zeng, Y., Zhang, P., Zhang, J., Lin, Z., Lu, H.: Towards high-resolution salient object detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 7234–7243 (2019) 14
54. Zhang, H., Zhang, J., Koniusz, P.: Few-shot learning via saliency-guided hallucination of samples. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2770–2779 (2019) 1
55. Zhang, J., Fan, D.P., Dai, Y., Anwar, S., Saleh, F.S., Zhang, T., Barnes, N.: Uc-net: uncertainty inspired rgb-d saliency detection via conditional variational autoencoders. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8582–8591 (2020) 14
56. Zhang, J., Yu, X., Li, A., Song, P., Liu, B., Dai, Y.: Weakly-supervised salient object detection via scribble annotations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 12546–12555 (2020) 4
57. Zhang, L., Yang, C., Lu, H., Ruan, X., Yang, M.H.: Ranking saliency. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(9), 1892–1904 (2016) 3
58. Zhang, L., Zhang, J., Lin, Z., Lu, H., He, Y.: Capsal: Leveraging captioning to boost semantics for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6024–6033 (2019) 4
59. Zhang, M., Ren, W., Piao, Y., Rong, Z., Lu, H.: Select, supplement and focus for rgb-d saliency detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3472–3481 (2020) 14
60. 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. pp. 714–722 (2018) 6
61. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image restoration. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020) 6
62. 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. pp. 3927–3936 (2019) 2, 4, 11, 12, 13
63. 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. pp. 8779–8788 (2019) 4