

EGNet: Edge Guidance Network for Salient Object Detection

Jia-Xing Zhao, Jiang-Jiang Liu, Deng-Ping Fan, Yang Cao, Ju-Feng Yang, Ming-Ming Cheng*
TKLNDST, CS, Nankai University

<http://mmcheng.net/egnet/>

Abstract

Fully convolutional neural networks (FCNs) have shown their advantages in the salient object detection task. However, most existing FCNs-based methods still suffer from coarse object boundaries. In this paper, to solve this problem, we focus on the complementarity between salient edge information and salient object information. Accordingly, we present an edge guidance network (EGNet) for salient object detection with three steps to simultaneously model these two kinds of complementary information in a single network. In the first step, we extract the salient object features by a progressive fusion way. In the second step, we integrate the local edge information and global location information to obtain the salient edge features. Finally, to sufficiently leverage these complementary features, we couple the same salient edge features with salient object features at various resolutions. Benefiting from the rich edge information and location information in salient edge features, the fused features can help locate salient objects, especially their boundaries more accurately. Experimental results demonstrate that the proposed method performs favorably against the state-of-the-art methods on six widely used datasets without any pre-processing and post-processing. The source code is available at <http://mmcheng.net/egnet/>.

1. Introduction

The goal of salient object detection (SOD) is to find the most visually distinctive objects in an image. It has received widespread attention recently and been widely used in many vision and image processing related areas, such as content-aware image editing [6], object recognition [42], photo-synth [4], non-photo-realist rendering [41], weakly supervised semantic segmentation [19] and image retrieval [15]. Besides, there are many works focusing on video salient object detection [12, 54] and RGB-D salient object detection [11, 66].

Inspired by cognitive studies of visual attention [7, 21,

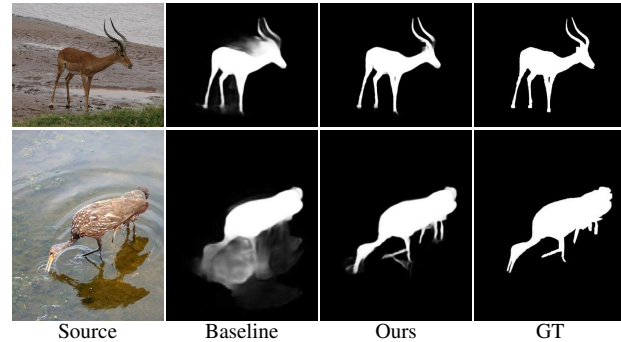


Figure 1. Visual examples of our method. After we model and fuse the salient edge information, the salient object boundaries become clearer.

[39], early works are mainly based on the fact that contrast plays the most important role in saliency detection. These methods benefit mostly from either global or local contrast cues and their learned fusion weights. Unfortunately, these hand-crafted features though can locate the most salient objects sometimes, the produced saliency maps are with irregular shapes because of the undesirable segmentation methods and unreliable when the contrast between the foreground and the background is inadequate.

Recently, convolutional neural networks (CNNs) [25] have successfully broken the limits of traditional hand-crafted features, especially after the emerging of Fully Convolutional Neural Networks (FCNs) [34]. These CNN-based methods have greatly refreshed the leaderboards on almost all the widely used benchmarks and are gradually replacing conventional salient object detection methods because of the efficiency as well as high performance. In SOD approaches based on CNNs architecture, the majority of them which regard the image patches [64, 65] as input use the multi-scale or multi-context information to obtain the final saliency map. Since the fully convolutional network is proposed for pixel labeling problems, several end-to-end deep architectures [17, 18, 23, 28, 31, 50, 60, 67] for salient object detection appear. The basic unit of output saliency map becomes per pixel from the image region. On the one hand, the result highlights the details because each pixel has its saliency value. However, on the other hand, it ignores the structure information which is important for SOD.

*M.M. Cheng (cmm@nankai.edu.cn) is the corresponding author.

With the increase of the network receptive field, the positioning of salient objects becomes more and more accurate. However, at the same time, spatial coherence is also ignored. Recently, to obtain the fine edge details, some SOD U-Net [40] based works [32, 33, 59, 61] used a bi-directional or recursive way to refine the high-level features with the local information. However, the boundaries of salient objects are still not explicitly modeled. The complementarity between the salient edge information and salient object information has not been noticed. Besides, there are some methods using pre-processing (Superpixel) [20] or post-processing (CRF) [17, 28, 33] to preserve the object boundaries. The main inconvenience with these approaches is their low inference speed.

In this paper, we focus on the complementarity between salient edge information and salient object information. We aim to leverage the salient edge features to help the salient object features locate objects, especially their boundaries more accurately. In summary, this paper makes three major contributions:

- We propose an EGNet to explicitly model complementary salient object information and salient edge information within the network to preserve the salient object boundaries. At the same time, the salient edge features are also helpful for localization.
- Our model jointly optimizes these two complementary tasks by allowing them to mutually help each other, which significantly improves the predicted saliency maps.
- We compare the proposed methods with 15 state-of-the-art approaches on six widely used datasets. Without bells and whistles, our method achieves the best performance under three evaluation metrics.

2. Related Works

Over the past years, some methods were proposed to detect the salient objects in an image. Early methods predicted the saliency map using a bottom-up pattern by the hand-craft feature, such as contrast [5], boundary background [57, 68], center prior [24, 44] and so on [22, 44, 51]. More details are introduced in [1, 2, 9].

Recently, Convolutional neural networks (CNNs) perform their advantages and refresh the state-of-the-art records in many fields of computer vision.

Li *et al.* [27] resized the image regions to three different scales to extract the multi-scale features and then aggregated these multiple saliency maps to obtain the final prediction map. Wang *et al.* [45] designed a neural network to extract the local estimation for the input patches and integrated these features with the global contrast and geometric information to describe the image patches. However, the result is limited by the performance of image patches in

these methods. In [34], long *et al.* firstly proposed a network (FCN) to predict the semantic label for each pixel. Inspired by FCN, more and more pixel-wise saliency detection methods were proposed. Wang *et al.* [47] proposed a recurrent FCN architecture for salient object detection. Hou *et al.* proposed a short connection [17, 18] based on HED [55] to integrate the low-level features and high-level features to solve the scale-space problem. In [62], Zhang *et al.* introduced a reformulated dropout and an effective hybrid upsampling to learn deep uncertain convolutional features to encourage robustness and accuracy. In [61], Zhang *et al.* explicitly aggregated the multi-level features into multiple resolutions and then combined these feature maps by a bidirectional aggregation method. Zhang *et al.* [59] proposed a bi-directional message-passing model to integrate multi-level features for salient object detection. Wang *et al.* [53] leveraged the fixation maps to help the model to locate the salient object more accurately. In [35], Luo *et al.* proposed a U-Net based architecture which contains an IOU edge loss to leverage the edge cues to detect the salient objects. In other saliency-related tasks, some methods of using edge cues have appeared. In [26], li *et al.* generated the contour of the object to obtain the salient instance segmentation results. In [29], li *et al.* leveraged the well-trained contour detection models to generate the saliency masks to overcome the limitation caused by manual annotations.

Compared with most of the SOD U-Net based methods [32, 33, 59, 61], we explicitly model edge information within the network to leverage the edge cues. Compared with the methods which use the edge cues [14, 58, 69], the major differences are that we use a single base network and jointly optimize the salient edge detection and the salient object detection, allowing them to help each other mutually, which results in better performance. Compared with NLDF [35], they implemented a loss function inspired by the Mumford-Shah function [38] to penalize errors on the edges. Since the salient edges are derived from salient objects through a fixed sober operator, this penalty essentially only affects the gradient in the neighborhood of salient edges on feature maps. In this way, the edge details are optimized to some extent, but the complementarity between salient edge detection and salient object detection is not sufficiently utilized. In our method, we design two modules to extract these two kinds of features independently. Then we fuse these complementary features by a one-to-one guidance module. In this way, the salient edge information can not only improve the quality of edges but also make the localization more accurate. The experimental part verifies our statement.

3. Salient Edge Guidance Network

The overall architecture is shown in Fig. 2. In this section, we begin by describing the motivations in Sec. 3.1, then introduce the adopted salient object feature extraction

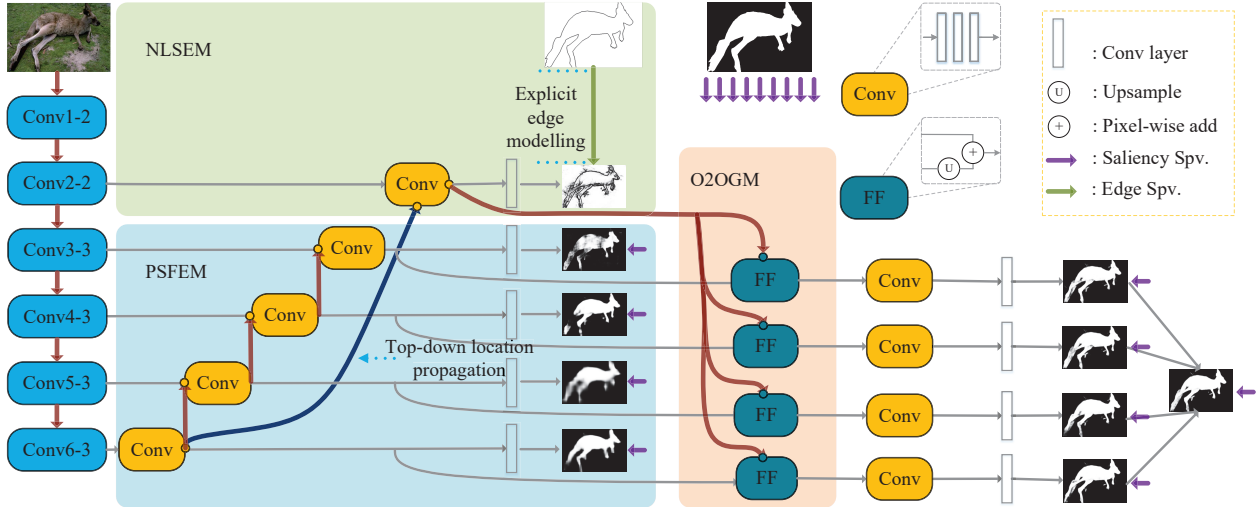


Figure 2. The pipeline of the proposed approach. We use brown thick lines to represent information flows between the scales. PSFEM: progressive salient object features extraction module. NLSEM: non-local salient edge features extraction module. O2OGM : one-to-one guidance module. FF: feature fusion. Spv.: supervision.

module and the proposed non-local salient edge features extraction module in Sec. 3.2, and finally introduce the proposed one-to-one guidance module in Sec. 3.3.

3.1. Motivation

The pixel-wise salient object detection methods have shown their advantages compared with region-based methods. However, they ignored the spatial coherence in the images, resulting in the unsatisfied salient object boundaries. Most methods [17, 18, 31, 33, 59, 61] hope to solve this problem by fusing multi-scale information. Some methods [17, 28, 33] used the post-processing such as CRF to refine the salient object boundaries. In NLDF [35], they proposed an IOU loss to affect the gradient of the location around the edge. None of them pay attention to the complementarity between salient edge detection and salient object detection. A good salient edge detection result can help salient object detection task in both segmentation and localization, and vice versa. Based on this idea, we proposed an EGNNet to model and fuse the complementary salient edge information and salient object information within a single network in an end-to-end manner.

3.2. Complementary information modeling

Our proposed network is independent of the backbone network. Here we use the VGG network suggested by other deep learning based methods [17, 35] to describe the proposed method. First, we truncate the last three fully connected layers. Following DSS [17, 18], we connect another side path to the last pooling layer in VGG. Thus from the backbone network, we obtain six side features Conv1-2, Conv2-2, Conv3-3, Conv4-3, Conv5-3, Conv6-3. Because the Conv1-2 is too close to the input and the receptive field is too small, we throw away this side path $S^{(1)}$. There are

five side paths $S^{(2)}, S^{(3)}, S^{(4)}, S^{(5)}, S^{(6)}$ remaining in our method. For simplicity, these five features could be denoted by a backbone features set C :

$$C = \{C^{(2)}, C^{(3)}, C^{(4)}, C^{(5)}, C^{(6)}\}, \quad (1)$$

where $C^{(2)}$ denotes the Conv2-2 features and so on. Conv2-2 preserves better edge information [61]. Thus we leverage the $S^{(2)}$ to extract the edge features and other side paths to extract the salient object features.

3.2.1 Progressive salient object features extraction

As shown in PSFEM of Fig. 2, to obtain richer context features, we leverage the widely used architecture U-Net [40] to generate the multi-resolution features. Different from the original U-Net, in order to obtain more robust salient object features, we add three convolutional layers (Conv in Fig. 2) on each side path, and after each convolutional layer, a ReLU layer is added to ensure the nonlinearity. To illustrate simply, we use the T (Tab. 1) to denote these convolutional layers and ReLU layers. Besides, deep supervision is used on each side path. We adopt a convolutional layer to convert the feature maps to the single-channel prediction mask and use D (Tab. 1) to denote it. The details of the convolutional layers could be found in Tab. 1.

3.2.2 Non-local salient edge features extraction

In this module, we aim to model the salient edge information and extract the salient edge features. As mentioned above, the Conv2-2 preserves better edge information. Hence we extract local edge information from Conv2-2. However, in order to get salient edge features, only local information is not enough. High-level semantic information or location information is also needed. When information

S	T_1				T_2				T_3				D		
2	3	1	128	3	1	128	3	1	128	3	1	1	3	1	1
3	3	1	256	3	1	256	3	1	256	3	1	1	3	1	1
4	5	2	512	5	2	512	5	2	512	5	2	1	3	1	1
5	5	2	512	5	2	512	5	2	512	5	2	1	3	1	1
6	7	3	512	7	3	512	7	3	512	7	3	1	3	1	1

Table 1. Details of each side output. T denotes the feature enhance module (Conv shown in Fig. 2). Each T contains three convolutional layers: T_1, T_2, T_3 and three followed ReLu layers. We show the kernel size, padding and channel number of each convolutional layer. For example, 3, 1, 128 denote a convolutional layer whose kernel size is 3, padding is 1, channel number is 128. D denotes the transition layer which converts the multi-channel feature map to one-channel activation map. S denotes the side path.

is progressively returned from the top level to the low level like the U-Net architecture, the high-level location information is gradually diluted. Besides, the receptive field of the top-level is the largest, and the location is the most accurate. Thus we design a top-down location propagation to propagate the top-level location information to the side path $S^{(2)}$ to restrain the non-salient edge. The fused features $\bar{C}^{(2)}$ could be denoted as:

$$\bar{C}^{(2)} = C^{(2)} + Up(\phi(Trans(\hat{F}^{(6)}; \theta)); C^{(2)}), \quad (2)$$

where $Trans(*; \theta)$ is a convolutional layer with parameter θ , which aims to change the number of channels of the feature, and $\phi()$ denotes a ReLU activation function. $Up(*; C^{(2)})$ is bilinear interpolation operation which aims to up-sample $*$ to the same size as $C^{(2)}$. On the right of the equation, the second term denotes the features from the higher side path. To illustrate clearly, we use $UpT(\hat{F}^{(i)}; \theta, C^{(j)})$ to represent $Up(\phi(Trans(\hat{F}^{(i)}; \theta)); C^{(j)})$. $\hat{F}^{(6)}$ denotes the enhanced features in side path $S^{(6)}$. The enhanced features $\hat{F}^{(6)}$ could be represented as $f(C^{(6)}; W_T^{(6)})$, and the enhanced features in $S^{(3)}, S^{(4)}, S^{(5)}$ could be computed as:

$$\hat{F}^{(i)} = f(C^{(i)} + UpT(\hat{F}^{(i+1)}; \theta, C^{(i)}); W_T^{(i)}), \quad (3)$$

where $W_T^{(i)}$ denotes the parameters in $T^{(i)}$ and $f(*; W_T^{(i)})$ denotes a series of convolutional and non-linear operations with parameters $W_T^{(i)}$.

After obtaining the guided features $\bar{C}^{(2)}$, similar with other side paths, we add a series convolutional layers to enhance the guided feature, then the final *salient edge features* F_E in $S^{(2)}$ could be computed as $f(\bar{C}^{(2)}; W_D^{(2)})$. The configuration details could be found in Tab. 1. To model the salient edge feature explicitly, we add an extra salient edge supervision to supervise the salient edge features. We use the cross-entropy loss which could be defined as:

$$\begin{aligned} \mathcal{L}^{(2)}(F_E; W_D^{(2)}) = & - \sum_{j \in Z_+} \log Pr(y_j = 1 | F_E; W_D^{(2)}) \\ & - \sum_{j \in Z_-} \log Pr(y_j = 0 | F_E; W_D^{(2)}), \end{aligned} \quad (4)$$

where Z_+ and Z_- denote the salient edge pixels set and background pixels set, respectively. W_D denotes the parameters of the transition layer as shown in Tab. 1. $Pr(y_j = 1 | F_E; W_D^{(2)})$ is the prediction map in which each value denotes the salient edge confidence for the pixel. In addition, the supervision added on the salient object detection side path can be represented as:

$$\begin{aligned} \mathcal{L}^{(i)}(\hat{F}^{(i)}; W_D^{(i)}) = & - \sum_{j \in Y_+} \log Pr(y_j = 1 | \hat{F}^{(i)}; W_D^{(i)}) \\ & - \sum_{j \in Y_-} \log Pr(y_j = 0 | \hat{F}^{(i)}; W_D^{(i)}), \quad i \in [3, 6], \end{aligned} \quad (5)$$

where Y_+ and Y_- denote the salient region pixels set and non-salient pixels set, respectively. Thus the total loss \mathbb{L} in the complementary information modeling could be denoted as:

$$\mathbb{L} = \mathcal{L}^{(2)}(F_E; W_D^{(2)}) + \sum_{i=3}^6 \mathcal{L}^{(i)}(\hat{F}^{(i)}; W_D^{(i)}). \quad (6)$$

3.3. One-to-one guidance module

After obtaining the complementary salient edge features and salient object features, we aim to leverage the salient edge features to guide the salient object features to perform better on both segmentation and localization. The simple way is to fuse the F_E and the $\hat{F}^{(3)}$. It will be better to sufficiently leverage the multi-resolution salient object features. However, the disadvantage of fusing the salient edge features and multi-resolution salient object features progressively from down to top is that salient edge features are diluted when salient object features are fused. Besides, the goal is to fuse salient object features and salient edge features to utilize complementary information to obtain better prediction results. Hence, we propose a one-to-one guidance module. Moreover, experimental parts validate our view.

Specifically, we add sub-side paths for $S^{(3)}, S^{(4)}, S^{(5)}, S^{(6)}$. In each sub-side path, by fusing the salient edge features into enhanced salient object features, we make the location of high-level predictions more accurate, and more importantly, the segmentation details become better. The salient edge guidance features (s-features) could be denoted as:

$$G^{(i)} = UpT(\hat{F}^{(i)}; \theta, F_E) + F_E, \quad i \in [3, 6]. \quad (7)$$

Then similar to the PSFEM, we adopt a series of convolutional layers T in each sub-side path to further enhance the s-features and a transition layer D to convert the multi-channel feature map to one-channel prediction map. Here in order to illustrate clearly, we denote the T and D as T' and D' in this module. By Eq. (3), we obtain the enhanced s-features $\hat{G}^{(i)}$.

Here we also add deep supervision for these enhanced s-features. For each sub-side output prediction map, the loss can be calculated as:

$$\mathcal{L}^{(i)'}(\hat{G}^{(i)}; W_{D'}^{(i)}) = - \sum_{j \in Y_+} \log Pr(y_j = 1 | \hat{G}^{(i)}; W_{D'}^{(i)}) - \sum_{j \in Y_-} \log Pr(y_j = 0 | \hat{G}^{(i)}; W_{D'}^{(i)}), \quad i \in [3, 6]. \quad (8)$$

Then we fuse the multi-scale refined prediction maps to obtain a fused map. The loss function for the fused map can be denoted as:

$$\mathcal{L}'_f(\hat{G}; W_{D'}) = \sigma(Y, \sum_{i=3}^6 \beta_i f(\hat{G}^{(i)}; W_{D'}^{(i)})), \quad (9)$$

where the $\sigma(*, *)$ represents the cross-entropy loss between prediction map and saliency ground-truth, which has the same form to Eq. (5). Thus the loss for this part and the total for the proposed network could be expressed as:

$$\mathbb{L}' = \mathcal{L}'_f(\hat{G}; W_{D'}) + \sum_{i=3}^6 \mathcal{L}^{(i)'}(\hat{G}^{(i)}; W_{D'}^{(i)}) \quad (10)$$

$$\mathbb{L}_t = \mathbb{L} + \mathbb{L}'.$$

4. Experiments

4.1. Implementation Details

We train our model on DUTS [46] dataset followed by [33, 49, 59, 63]. For a fair comparison, we use VGG [43] and ResNet [16] as backbone networks, respectively. Our model is implemented in PyTorch. All the weights of newly added convolution layers are initialized randomly with a truncated normal ($\sigma = 0.01$), and the biases are initialized to 0. The hyper-parameters are set as followed: learning rate = $5e-5$, weight decay = 0.0005, momentum = 0.9, loss weight for each side output is equal to 1. A back propagation is processing for each of the ten images. We do not use the validation dataset during training. We train our model 24 epochs and divide the learning rate by 10 after 15 epochs. During inference, we are able to obtain a predicted salient edge map and a set of saliency maps. In our method, we directly use the fused prediction map as the final saliency map.

4.2. Datasets and Evaluation Metric

We have evaluated the proposed architecture on six widely used public benchmark datasets: ECSSD [56], PASCAL-S [30], DUT-OMRON [57], SOD [36, 44], HKU-IS [27], DUTS [46]. ECSSD [56] contains 1000 meaningful semantic images with various complex scenes. PASCAL-S [30] contains 850 images which are chosen from the validation set of the PASCAL VOC segmentation dataset [8].

DUT-OMRON [57] contains 5168 high-quality but challenging images. Images in this dataset contain one or more salient objects with a relatively complex background. SOD [36] contains 300 images and is proposed for image segmentation. Pixel-wise annotations of salient objects are generated by [44]. It is one of the most challenging datasets currently. HKU-IS [27] contains 4447 images with high-quality annotations, many of which have multiple disconnected salient objects. This dataset is split into 2500 training images, 500 validation images and 2000 test images. DUTS [46] is the largest salient object detection benchmark. It contains 10553 images for training and 5019 images for testing. Most images are challenging with various locations and scales. Following most recent works [33, 49, 52], we use the DUTS dataset to train the proposed model.

We use three widely used and standard metrics, F-measure, mean absolute error (MAE) [2], and a recently proposed structure-based metric, namely S-measure [10], to evaluate our model and other state-of-the-art models. F-measure is a harmonic mean of average precision and average recall, formulated as:

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

we set $\beta^2 = 0.3$ to weigh precision more than recall as suggested in [5]. Precision denotes the ratio of detected salient pixels in the predicted saliency map. Recall denotes the ratio of detected salient pixels in the ground-truth map. Precision and recall are computed on binary images. Thus we should threshold the prediction map to binary map first. There are different precision and recall of different thresholds. We could plot the precision-recall curve at different thresholds. Here we use the code provided by [17, 18] for evaluation. Following most salient object detection methods [17, 18, 32, 59], we report the maximum F-measure from all precision-recall pairs.

MAE is a metric which evaluates the average difference between prediction map and ground-truth map. Let P and Y denote the saliency map and the ground truth that is normalized to $[0, 1]$. We compute the MAE score by:

$$\varepsilon = \frac{1}{W \times H} \sum_{x=1}^W \sum_{y=1}^H |P(x, y) - Y(x, y)|, \quad (12)$$

where W and H are the width and height of images, respectively.

S-measure focuses on evaluating the structural information of saliency maps, which is closer to the human visual system than F-measure. Thus we include S-measure for a more comprehensive evaluation. S-measure could be computed as:

$$S = \gamma S_o + (1 - \gamma) S_r, \quad (13)$$

	ECSSD [56]			PASCAL-S [30]			DUT-O [57]			HKU-IS [27]			SOD [36, 37]			DUTS-TE [46]		
	MaxF ↑	MAE ↓	S ↑	MaxF ↑	MAE ↓	S ↑	MaxF ↑	MAE ↓	S ↑	MaxF ↑	MAE ↓	S ↑	MaxF ↑	MAE ↓	S ↑	MaxF ↑	MAE ↓	S ↑
VGG-based																		
DCL* [28]	0.896	0.080	0.863	0.805	0.115	0.791	0.733	0.094	0.743	0.893	0.063	0.859	0.831	0.131	0.748	0.786	0.081	0.785
DSS* [17, 18]	0.906	0.064	0.882	0.821	0.101	0.796	0.760	0.074	0.765	0.900	0.050	0.878	0.834	0.125	0.744	0.813	0.065	0.812
MSR [26]	0.903	0.059	0.875	0.839	0.083	0.802	0.790	0.073	0.767	0.907	0.043	0.852	0.841	0.111	0.757	0.824	0.062	0.809
NLDF [35]	0.903	0.065	0.875	0.822	0.098	0.803	0.753	0.079	0.750	0.902	0.048	0.878	0.837	0.123	0.756	0.816	0.065	0.805
RAS [3]	0.915	0.060	0.886	0.830	0.102	0.798	0.784	0.063	0.792	0.910	0.047	0.884	0.844	0.130	0.760	0.800	0.060	0.827
ELD* [13]	0.865	0.082	0.839	0.772	0.122	0.757	0.738	0.093	0.743	0.843	0.072	0.823	0.762	0.154	0.705	0.747	0.092	0.749
DHS [32]	0.905	0.062	0.884	0.825	0.092	0.807	-	-	-	0.892	0.052	0.869	0.823	0.128	0.750	0.815	0.065	0.809
RFCN* [48]	0.898	0.097	0.852	0.827	0.118	0.799	0.747	0.094	0.752	0.895	0.079	0.860	0.805	0.161	0.730	0.786	0.090	0.784
UCF [62]	0.908	0.080	0.884	0.820	0.127	0.806	0.735	0.131	0.748	0.888	0.073	0.874	0.798	0.164	0.762	0.771	0.116	0.777
Amulet [61]	0.911	0.062	0.894	0.826	0.092	0.820	0.737	0.083	0.771	0.889	0.052	0.886	0.799	0.146	0.753	0.773	0.075	0.796
C2S [29]	0.909	0.057	0.891	0.845	0.081	0.839	0.759	0.072	0.783	0.897	0.047	0.886	0.821	0.122	0.763	0.811	0.062	0.822
PAGR [63]	0.924	0.064	0.889	0.847	0.089	0.818	0.771	0.071	0.751	0.919	0.047	0.889	0.841	0.146	0.716	0.854	0.055	0.825
Ours	0.941	0.044	0.913	0.863	0.076	0.848	0.826	0.056	0.813	0.929	0.034	0.910	0.869	0.110	0.788	0.880	0.043	0.866
ResNet-based																		
SRM* [49]	0.916	0.056	0.895	0.838	0.084	0.832	0.769	0.069	0.777	0.906	0.046	0.887	0.840	0.126	0.742	0.826	0.058	0.824
DGRL [52]	0.921	0.043	0.906	0.844	0.075	0.839	0.774	0.062	0.791	0.910	0.036	0.896	0.843	0.103	0.774	0.828	0.049	0.836
PICANet* [33]	0.932	0.048	0.914	0.864	0.077	0.850	0.820	0.064	0.808	0.920	0.044	0.905	0.861	0.103	0.790	0.863	0.050	0.850
Ours	0.943	0.041	0.918	0.869	0.074	0.852	0.842	0.052	0.818	0.937	0.031	0.918	0.890	0.097	0.807	0.893	0.039	0.875

Table 2. Quantitative comparison including max F-measure, MAE, and S-measure over six widely used datasets. ‘-’ denotes that corresponding methods are trained on that dataset. ↑ & ↓ denote larger and smaller is better, respectively. * means methods using pre-processing or post-processing. The best three results are marked in red, blue, and green, respectively. Our method achieves the state-of-the-art on these six widely used datasets under three evaluation metrics.

Model	SOD			DUTS		
	MaxF ↑	MAE ↓	S ↑	MaxF ↑	MAE ↓	S ↑
1. B	.851	.116	.780	.855	.060	.844
2. B + edge_PROG	.873	.105	.799	.872	.051	.851
3. B + edge_TDLP	.882	.100	.807	.879	.044	.866
4. B + edge_NLDF	.857	.112	.794	.866	.053	.860
5. B + edge_TDLP + MRF_PROG	.882	.106	.796	.880	.046	.869
6. B + edge_TDLP + MRF_OTO	.890	.097	.807	.893	.039	.875

Table 3. Ablation analyses on SOD [36] and DUTS-TE [46]. Here, B denotes the baseline model. edge_PROG, edge_TDLP, edge_NLDF, MRF_PROG, MRF_OTO are introduced in the Sec. 4.3.

where S_o and S_r denotes the region-aware and object-aware structural similarity and γ is set as 0.5 by default. More details can be found in [10].

4.3. Ablation Experiments and Analyses

In this section, with the DUTS-TR [46] as the training set, we explore the effect of different components in the proposed network over the relatively difficult dataset SOD [36] and the recently proposed big dataset DUTS-TE [46].

4.3.1 The complementary information modeling

In this subsection, we explore the role of salient edge information, which is also our basic idea. The baseline is the U-Net architecture which integrates the multi-scale features (From Conv2-2 to Conv6-3) in the way as PSFEM (Fig. 2). We remove the side path $S^{(2)}$ in the baseline and then fuse the final saliency features $\hat{F}^{(3)}$ (side path from Conv3-3) and the local Conv2-2 features to obtain the salient edge features. Finally, we integrate salient edge features and the

salient object features $\hat{F}^{(3)}$ to get the prediction mask. We denote this strategy of using edges as edge_PROG. The result is shown in the second row of Tab. 3. It proves that the salient edge information is very useful for the salient object detection task.

4.3.2 Top-down location propagation

In this subsection, we explore the role of top-down location propagation. Compared with edge_PROG mentioned in the previous subsection Sec. 4.3.1, we leverage the top-down location propagation to extract more accurate location information from top-level instead of side path $S^{(3)}$. We call this strategy of using edges as edge_TDLP. By comparing the second and third rows of Tab. 3, the effect of top-down location propagation could be proved. Besides, comparing the first row and the third row of Tab. 3, we can find that through our explicit modeling of these two kinds of complementary information within the network, the performance is greatly improved on the datasets (3.1%, 2.4% under F-measure) without additional time and space consumption.

4.3.3 Mechanism of using edge cues

To demonstrate the advantages over NLDF [35], in which an IOU loss is added to the end of the network to punish the errors of edges. We add the same IOU loss to the baseline. This strategy is called edge_NLDF. The performance is shown in the 4th row of Tab. 3. Compared with the baseline model, the improvement is limited. This also demonstrates that the proposed method of using edge information is more

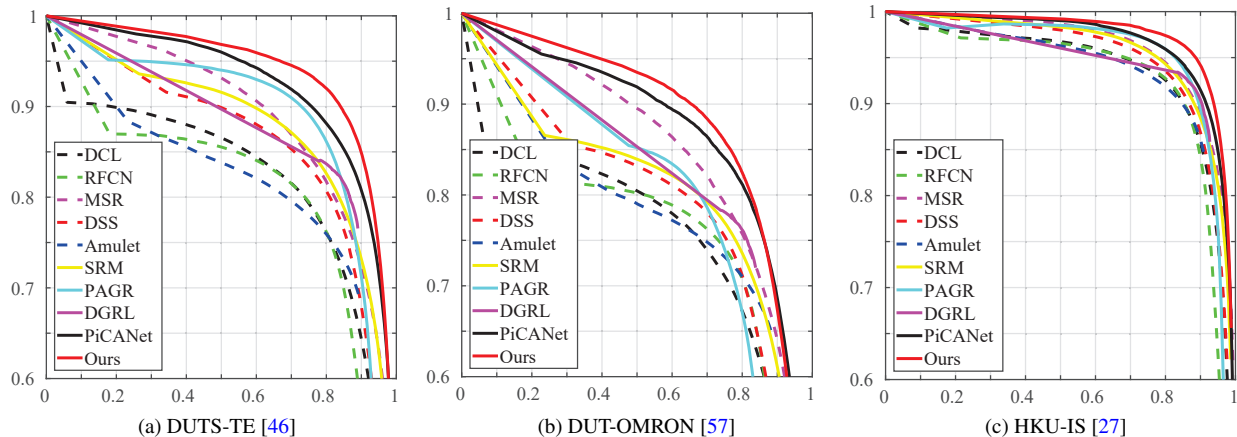


Figure 3. Precision (vertical axis) recall (horizontal axis) curves on three popular salient object datasets. It can be seen that the proposed method performs favorably against state-of-the-arts.

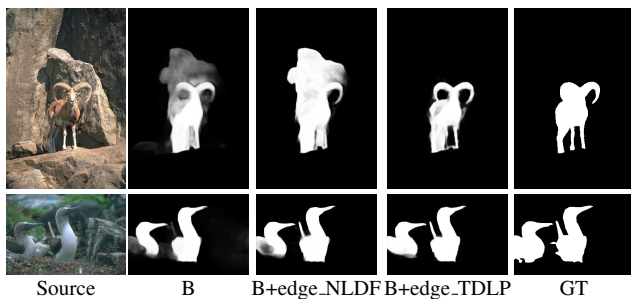


Figure 4. Visual examples before and after adding edge cues. B denotes the baseline model. edge_NLDF and edge_TDLP represent the edges penalty used in NLDF [35] and the edge modeling method proposed in this paper. The details are introduced in the Sec. 4.3.

effective. The visualization results are shown in Fig. 4. Compared with the baseline model without edge constraint, after we add the edge penalty used in NLDF [35], edge information can only help refine the boundaries. In particular, this penalty can not help to remove the redundant parts in saliency prediction mask, nor can it make up for the missing parts. In contrast, the proposed complementary information modeling method considers the complementarity between salient edge information and salient object information, and performs better on both segmentation and localization.

Besides, in order to further prove that salient edge detection and salient object detection are mutually helpful and complementary. We compare the salient edges generated by NLDF with the salient edges generated by us. The pre-trained model and code are both provided by the authors. As shown in Tab. 4, it could be found that the salient edge generated by our method is much better, especially under the recall and F-measure metrics. It proves that the edges are more accurate in our methods.

Model	SOD			DUTS		
	Recall \uparrow	Precision \uparrow	MaxF \uparrow	Recall \uparrow	Precision \uparrow	MaxF \uparrow
NLDF	0.513	0.541	0.527	0.318	0.659	0.429
Ours	0.637	0.534	0.581	0.446	0.680	0.539

Table 4. Comparisons on the salient edge generated by the NLDF and ours.

4.3.4 The complementary features fusion

After we obtain the salient edge features and multi-resolution salient object features. We aim to fuse these complementary features. Here we compare three fusion methods. The first way is the default way, which integrates the salient edge features (F_E) and the salient object features $\hat{F}^{(3)}$ which is on the top of U-Net architecture. The second way is to fuse the multi-resolution features $\hat{F}^{(3)}$, $\hat{F}^{(4)}$, $\hat{F}^{(5)}$, $\hat{F}^{(6)}$ progressively, which is called MRF_PROG. The third way is the proposed one-to-one guidance, which is denoted MRF_OTO. Here MRF denotes the multi-resolution fusion. The results are shown in the third, fifth, sixth rows of Tab. 3, respectively. It can be seen that our proposed one-to-one guidance method is most suitable for our whole architecture.

4.4. Comparison with the State-of-the-art

In this section, we compare our proposed EGNet with 15 previous state-of-the-art methods, including DCL [28], DSS [17, 18], NLDF [35], MSR [26], ELD [13], DHS [32], RFCN [48], UCF [62], Amulet [61], PAGR [63], PiCANet [33], SRM [49], DGRL [52], RAS [3] and C2S [29]. Note that all the saliency maps of the above methods are produced by running source codes or pre-computed by the authors. The evaluation codes are provided in [10, 17, 18].

F-measure, MAE, and S-measure. We evaluate and compare our proposed method with other salient object detection methods in term of F-measure, MAE, and S-measure as shown in Tab. 2. We could see that different methods

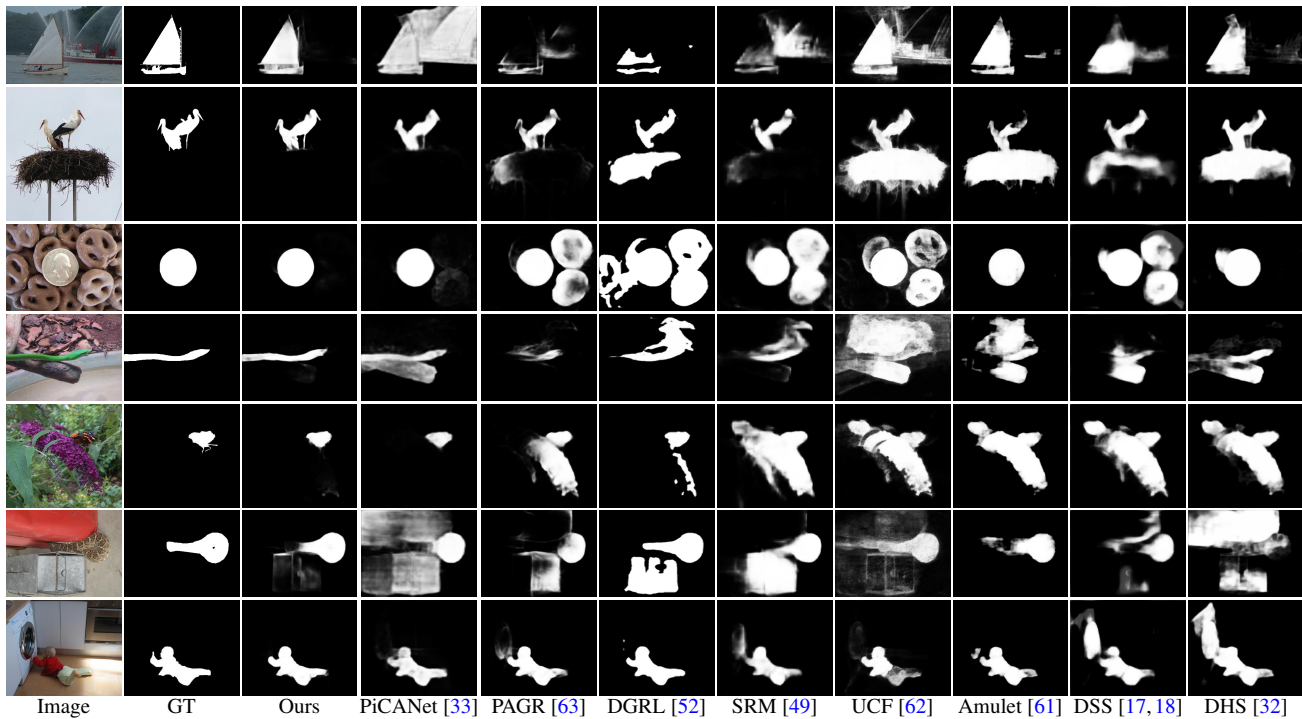


Figure 5. Qualitative comparisons with state-of-the-arts.

may use different backbone net. Here for a fair comparison, we train our model on the VGG [43] and ResNet [16], respectively. It can be seen that our model performs favorably against the state-of-the-art methods under all evaluation metrics on all the compared datasets especially on the relative challenging dataset SOD [36, 44] (2.9% and 1.7% improvements in F-measure and S-measure) and the largest dataset DUTS [46] (3.0% and 2.5%). Specifically, Compared with the current best approach [33], the average F-measure improvement on six datasets is 1.9%. Note that this is achieved without any pre-processing and post-processing.

Precision-recall curves. Besides the numerical comparisons shown in Tab. 2, we plot the precision-recall curves of all compared methods over three datasets Fig. 3. As can be seen that the solid red line which denotes the proposed method outperforms all other methods at most thresholds. Due to the help of the complementary salient edge information, the results yield sharp edge information and accurate localization, which results in a better PR curve.

Visual comparison. In Fig. 5, we show some visualization results. It could be seen that our method performs better on salient object segmentation and localization. It is worth mentioning that thank to the salient edge features, our result could not only highlight the salient region but also produce coherent edges. For instance, for the first sample, due to the influence of the complex scene, other methods are not capable of localizing and segmenting salient objects accurately. However, benefiting from the complementary salient edge features, our method performs better. For the second sam-

ple, in which the salient object is relatively small, our result is still very close to the ground-truth.

5. Conclusion

In this paper, we aim to preserve salient object boundaries well. Different from other methods which integrate the multi-scale features or leverage the post-processing, we focus on the complementarity between salient edge information and salient object information. Based on this idea, we propose the EGNet to model these complementary features within the network. First, we extract the multi-resolution salient object features based on U-Net. Then, we propose a non-local salient edge features extraction module which integrates the local edge information and global location information to get the salient edge features. Finally, we adopt a one-to-one guidance module to fuse these complementary features. The salient object boundaries and localization are improved under the help of salient edge features. Our model performs favorably against the state-of-the-art methods on six widely used datasets without any pre-processing or post-processing. We also provide analyses of the effectiveness of the EGNet.

Acknowledgments. This research was supported by NSFC (61572264), the national youth talent support program, and Tianjin Natural Science Foundation (17JCJQC43700, 18ZXZNGX00110).

References

- [1] Ali Borji, Ming-Ming Cheng, Qibin Hou, Huaizu Jiang, and Jia Li. Salient object detection: A survey. *CVM*, 5(2):117–150, 2019.
- [2] Ali Borji, Ming-Ming Cheng, Huaizu Jiang, and Jia Li. Salient object detection: A benchmark. *IEEE TIP*, 24(12):5706–5722, 2015.
- [3] Shuhan Chen, Xiuli Tan, Ben Wang, and Xuelong Hu. Reverse attention for salient object detection. In *ECCV*, pages 234–250, 2018.
- [4] Tao Chen, Ming-Ming Cheng, Ping Tan, Ariel Shamir, and Shi-Min Hu. Sketch2photo: Internet image montage. *ACM TOG*, 28(5):124:1–10, 2009.
- [5] Ming Cheng, Niloy J Mitra, Xumin Huang, Philip HS Torr, and Song Hu. Global contrast based salient region detection. *IEEE TPAMI*, 37(3):569–582, 2015.
- [6] Ming-Ming Cheng, Fang-Lue Zhang, Niloy J Mitra, Xiaolei Huang, and Shi-Min Hu. Refinder: finding approximately repeated scene elements for image editing. *ACM TOG*, 29(4):83, 2010.
- [7] Wolfgang Einhäuser and Peter König. Does luminance-contrast contribute to a saliency map for overt visual attention? *European Journal of Neuroscience*, 17(5):1089–1097, 2003.
- [8] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010.
- [9] Deng-Ping Fan, Ming-Ming Cheng, Jiang-Jiang Liu, Shang-Hua Gao, Qibin Hou, and Ali Borji. Salient objects in clutter: Bringing salient object detection to the foreground. In *ECCV*, pages 186–202. Springer, 2018.
- [10] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *ICCV*, pages 4548–4557, 2017.
- [11] Deng-Ping Fan, Zheng Lin, Jia-Xing Zhao, Yun Liu, Zhao Zhang, Qibin Hou, Menglong Zhu, and Ming-Ming Cheng. Rethinking rgb-d salient object detection: Models, datasets, and large-scale benchmarks. *arXiv preprint arXiv:1907.06781*, 2019.
- [12] Deng-Ping Fan, Wenguan Wang, Ming-Ming Cheng, and Jianbing Shen. Shifting more attention to video salient object detection. In *CVPR*, pages 8554–8564, 2019.
- [13] Lee Gayoung, Tai Yu-Wing, and Kim Junmo. Deep saliency with encoded low level distance map and high level features. In *CVPR*, 2016.
- [14] Wenlong Guan, Tiantian Wang, Jinqing Qi, Lihe Zhang, and Huchuan Lu. Edge-aware convolution neural network based salient object detection. *IEEE SPL*, 26(1):114–118, 2018.
- [15] Junfeng He, Jinyuan Feng, Xianglong Liu, Tao Cheng, Tai-Hsu Lin, Hyunjin Chung, and Shih-Fu Chang. Mobile product search with bag of hash bits and boundary reranking. In *CVPR*, pages 3005–3012, 2012.
- [16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *ICCV*, pages 770–778, 2016.
- [17] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip Torr. Deeply supervised salient object detection with short connections. In *CVPR*, pages 3203–3212, 2017.
- [18] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip Torr. Deeply supervised salient object detection with short connections. *IEEE TPAMI*, 41(4):815–828, 2019.
- [19] Qibin Hou, Peng-Tao Jiang, Yunchao Wei, and Ming-Ming Cheng. Self-erasing network for integral object attention. In *NIPS*, 2018.
- [20] Ping Hu, Bing Shuai, Jun Liu, and Gang Wang. Deep level sets for salient object detection. In *CVPR*, pages 2300–2309, 2017.
- [21] Laurent Itti and Christof Koch. Computational modeling of visual attention. *Nature reviews neuroscience*, 2(3):194–203, 2001.
- [22] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE TPAMI*, 20(11):1254–1259, 1998.
- [23] Sen Jia and Neil DB Bruce. Richer and deeper supervision network for salient object detection. *arXiv preprint arXiv:1901.02425*, 2019.
- [24] Dominik A Klein and Simone Frintrop. Center-surround divergence of feature statistics for salient object detection. In *ICCV*, pages 2214–2219. IEEE, 2011.
- [25] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [26] Guanbin Li, Yuan Xie, Liang Lin, and Yizhou Yu. Instance-level salient object segmentation. In *CVPR*, 2017.
- [27] Guanbin Li and Yizhou Yu. Visual saliency based on multi-scale deep features. In *CVPR*, pages 5455–5463, 2015.
- [28] Guanbin Li and Yizhou Yu. Deep contrast learning for salient object detection. In *CVPR*, 2016.
- [29] Xin Li, Fan Yang, Hong Cheng, Wei Liu, and Dinggang Shen. Contour knowledge transfer for salient object detection. In *ECCV*, pages 355–370, 2018.
- [30] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *CVPR*, pages 280–287, 2014.
- [31] Zun Li, Congyan Lang, Yunpeng Chen, Junhao Liew, and Jiashi Feng. Deep reasoning with multi-scale context for salient object detection. *arXiv preprint arXiv:1901.08362*, 2019.
- [32] Nian Liu and Junwei Han. Dhsnet: Deep hierarchical saliency network for salient object detection. In *CVPR*, pages 678–686, 2016.
- [33] Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Learning pixel-wise contextual attention for saliency detection. In *CVPR*, pages 3089–3098, 2018.
- [34] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015.
- [35] Zhiming Luo, Akshaya Kumar Mishra, Andrew Achkar, Justin A Eichel, Shaozi Li, and Pierre-Marc Jodoin. Non-local deep features for salient object detection. In *CVPR*, 2017.

- [36] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, volume 2, pages 416–423, 2001.
- [37] Vida Movahedi and James H Elder. Design and perceptual validation of performance measures for salient object segmentation. In *IEEE CVPRW*, pages 49–56. IEEE, 2010.
- [38] David Mumford and Jayant Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *CPAM*, 42(5):577–685, 1989.
- [39] Derrick Parkhurst, Klinton Law, and Ernst Niebur. Modeling the role of salience in the allocation of overt visual attention. *Vision research*, 42(1):107–123, 2002.
- [40] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [41] Paul L Rosin and Yu-Kun Lai. Artistic minimal rendering with lines and blocks. *Graphical Models*, 75(4):208–229, 2013.
- [42] Ueli Rutishauser, Dirk Walther, Christof Koch, and Pietro Perona. Is bottom-up attention useful for object recognition? In *CVPR*, 2004.
- [43] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.
- [44] Jingdong Wang, Huaizu Jiang, Zejian Yuan, Ming-Ming Cheng, Xiaowei Hu, and Nanning Zheng. Salient object detection: A discriminative regional feature integration approach. *IJCV*, 123(2):251–268, 2017.
- [45] Lijun Wang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Deep networks for saliency detection via local estimation and global search. In *ICCV*, pages 3183–3192, 2015.
- [46] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *CVPR*, pages 136–145, 2017.
- [47] Linzhao Wang, Lijun Wang, Huchuan Lu, Pingping Zhang, and Xiang Ruan. Saliency detection with recurrent fully convolutional networks. In *ECCV*, pages 825–841. Springer, 2016.
- [48] Linzhao Wang, Lijun Wang, Huchuan Lu, Pingping Zhang, and Xiang Ruan. Saliency detection with recurrent fully convolutional networks. In *ECCV*, 2016.
- [49] Tiantian Wang, Ali Borji, Lihe Zhang, Pingping Zhang, and Huchuan Lu. A stagewise refinement model for detecting salient objects in images. In *ICCV*, pages 4019–4028, 2017.
- [50] Tiantian Wang, Yongri Piao, Li Xiao, Lihe Zhang, and Huchuan Lu. Deep learning for light field saliency detection. In *ICCV*, 2019.
- [51] Tiantian Wang, Lihe Zhang, Huchuan Lu, Chong Sun, and Jinqing Qi. Kernelized subspace ranking for saliency detection. In *ECCV*, pages 450–466, 2016.
- [52] Tiantian Wang, Lihe Zhang, Shuo Wang, Huchuan Lu, Gang Yang, Xiang Ruan, and Ali Borji. Detect globally, refine locally: A novel approach to saliency detection. In *CVPR*, pages 3127–3135, 2018.
- [53] Wenguan Wang, Jianbing Shen, Xingping Dong, and Ali Borji. Salient object detection driven by fixation prediction. In *ICCV*, pages 1711–1720, 2018.
- [54] Ziqin Wang, Jun Xu, Li Liu, Fan Zhu, and Ling Shao. Ranet: Ranking attention network for fast video object segmentation. In *ICCV*, Oct 2019.
- [55] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *ICCV*, pages 1395–1403, 2015.
- [56] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *CVPR*, pages 1155–1162, 2013.
- [57] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, pages 3166–3173, 2013.
- [58] Jing Zhang, Yuchao Dai, Fatih Porikli, and Mingyi He. Deep edge-aware saliency detection. *arXiv preprint arXiv:1708.04366*, 2017.
- [59] Lu Zhang, Ju Dai, Huchuan Lu, You He, and Gang Wang. A bi-directional message passing model for salient object detection. In *ICCV*, pages 1741–1750, 2018.
- [60] Pingping Zhang, Wei Liu, Huchuan Lu, and Chunhua Shen. Salient object detection with lossless feature reflection and weighted structural loss. *IEEE TIP*, 2019.
- [61] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Xiang Ruan. Amulet: Aggregating multi-level convolutional features for salient object detection. In *ICCV*, pages 202–211, 2017.
- [62] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Baocai Yin. Learning uncertain convolutional features for accurate saliency detection. In *ICCV*, pages 212–221. IEEE, 2017.
- [63] Xiaoning Zhang, Tiantian Wang, Jinqing Qi, Huchuan Lu, and Gang Wang. Progressive attention guided recurrent network for salient object detection. In *CVPR*, pages 714–722, 2018.
- [64] Jiaying Zhao, Ren Bo, Qibin Hou, Ming-Ming Cheng, and Paul Rosin. Flic: Fast linear iterative clustering with active search. *CVM*, 4(4):333–348, Dec 2018.
- [65] Jiaying Zhao, Bo Ren, Qibin Hou, and Ming-Ming Cheng. Flic: Fast linear iterative clustering with active search. In *AAAI*, 2018.
- [66] Jia-Xing Zhao, Yang Cao, Deng-Ping Fan, Ming-Ming Cheng, Xuan-Yi Li, and Le Zhang. Contrast prior and fluid pyramid integration for rgbd salient object detection. In *CVPR*, 2019.
- [67] Kai Zhao, Shanghua Gao, Wenguan Wang, and Ming-Ming Cheng. Optimizing the f-measure for threshold-free salient object detection. In *ICCV*, Oct 2019.
- [68] Wangjiang Zhu, Shuang Liang, Yichen Wei, and Jian Sun. Saliency optimization from robust background detection. In *CVPR*, pages 2814–2821, 2014.
- [69] Yunzhi Zhuge, Gang Yang, Pingping Zhang, and Huchuan Lu. Boundary-guided feature aggregation network for salient object detection. *IEEE SPL*, 25(12):1800–1804, 2018.