# A Bi-directional Message Passing Model for Salient Object Detection

Lu Zhang[1], Ju Dai[1], Huchuan Lu[1], You He[2], Gang Wang[3]
[1]Dalian University of Technology, China
[2]Naval Aviation University, China
[3]Alibaba AILabs, China
{luzhang_dut, daijucug}@mail.dlut.edu.cn, lhchuan@dlut.edu.cn,
heyou_f@126.com, wg134231@alibaba-inc.com

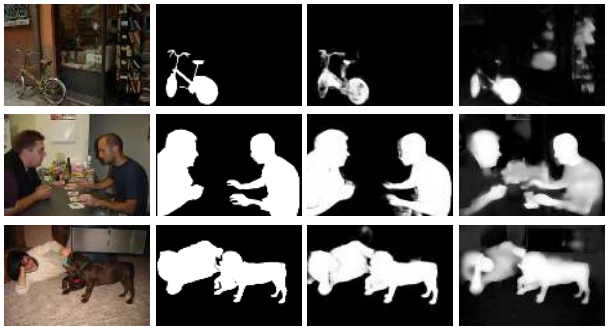
## Abstract

*Recent progress on salient object detection is beneficial from Fully Convolutional Neural Network (FCN). The saliency cues contained in multi-level convolutional features are complementary for detecting salient objects. How to integrate multi-level features becomes an open problem in saliency detection. In this paper, we propose a novel bi-directional message passing model to integrate multi-level features for salient object detection. At first, we adopt a Multi-scale Context-aware Feature Extraction Module (MCFEM) for multi-level feature maps to capture rich context information. Then a bi-directional structure is designed to pass messages between multi-level features, and a gate function is exploited to control the message passing rate. We use the features after message passing, which simultaneously encode semantic information and spatial details, to predict saliency maps. Finally, the predicted results are efficiently combined to generate the final saliency map. Quantitative and qualitative experiments on five benchmark datasets demonstrate that our proposed model performs favorably against the state-of-the-art methods under different evaluation metrics.*

## 1. Introduction

Salient object detection aims to localize the most conspicuous and eye-attracting regions in an image. It can be taken as a pre-processing step in many computer vision tasks, such as scene classification [25], visual tracking [3, 21], person re-identification [40] and image retrieval [5]. Although numerous valuable models have been proposed, it is still difficult to locate salient object accurately especially in some complicated scenarios.

In recent years, the Fully Convolutional Neural Network (FCN) has shown impressive results in dense prediction tasks, such as semantic segmentation [19], contour detection [31] and pose estimation [34]. Motivated by this, several attempts have been performed to utilize FCN in salient object detection [18, 38, 28, 20, 37]. Although these models could achieve promising results, there are still two problems. Firstly, most previous FCN-based saliency detection models [29, 27, 38, 12] stack single-scale convolutional and max pooling layers sequentially to generate deep features. Due to the limited receptive fields, the learned features might not contain rich context information to precisely detect the objects with various scales, shapes and locations (see the fourth column in Fig. 1). Secondly, early works [29, 18, 15, 12] predict saliency maps by mainly using high-level features from deep convolutional layers. The lack of low-level spatial details may make the saliency maps fail to retain fine object boundaries. This motivates several efforts [37, 8, 14] to exploit multi-level convolutional features for saliency detection. Hou *et al.* [8] propose to add short connections between multiple side output layers to combine features of different levels. However, the short connections are only performed from deep side output layers to shallow ones and ignore the information transmission in the opposite direction. Thus the deep side outputs still lack the low-level details contained in shallow side output



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

Figure 1. Visual examples of our method and RFCN [29]. From left to right: (a) input image, (b) ground truth, (c) saliency map of the proposed method, (d) saliency map of RFCN [29].

layers. Another work by Zhang *et al.* [37] aggregates multi-level features by concatenating feature maps from both high level and low level. However, the direct concatenation of feature maps at all levels without weighting their importance is not the optimal way to effectively fuse them. As the multi-level features are not always useful for every input image, this aggregation method would lead to information redundancy. More importantly, inaccurate information at some levels would lead to a performance degradation or even wrong prediction. In consequence, it is of great importance to design a mechanism to filter the undesired features and let the beneficial ones at each level adaptively fuse with other levels.

In this paper, we propose a novel bi-directional message passing model for salient object detection. For the first problem, we design a Multi-scale Context-aware Feature Extraction Module (MCFEM) to capture multi-scale contextual information. For each side output, we obtain multiple feature maps by stacking dilated convolutional layers [35] with different receptive fields. The feature maps are then fused by concatenating to capture objects as well as useful image context at multiple scales. For the second problem, we introduce a Gated Bi-directional Message Passing Module (GBMPM). We put forward a bi-directional structure to pass messages among features from different levels. With this structure, high-level semantic information in deeper layers is passed to shallower layers and low-level spatial details contained in shallower layers are passed in the opposite direction. As a result, the semantic concept and fine details are incorporated at each level. Besides, we use the gate function to control the message passing, so that the useful features are transmitted and the superfluous features are abandoned. The GBMPM provides an adaptive and effective strategy for incorporating multi-level features. The integrated features are complementary with each other and robust for handling different scenes. In summary, the MCFEM and GBMPM in our model work collaboratively to accurately detect the salient objects (see the third column in Fig. 1). Our contributions are summarized as three folds:

- We propose a multi-scale context-aware feature extraction module to capture rich context information for multi-level features to localize salient objects with various scales.

- We put forward a gated bi-directional message passing module to adaptively and effectively incorporate multi-level convolutional features. The integrated features are complementary and robust for detecting salient objects in various scenes.

- We compare the proposed approach with 13 state-of-the-art saliency detection methods on five datasets. Our method achieves the best performance under different evaluation metrics. Besides, the proposed model has a near real-time speed of 22 fps.

## 2. Related Work

### 2.1. Salient Object Detection

Early methods predict saliency using bottom-up computational models and low-level hand-crafted features. A majority of them utilize heuristic saliency priors, such as color contrast [4, 1], boundary background [33, 41] and center prior [10]. More details about the traditional methods could be referred in [2].

In recent years, convolutional neural network (CNN) has achieved competitive performance in many computer vision tasks. In salient object detection, a lot of deep learning models with various network architectures have been proposed. Some early deep saliency models utilize CNN features to predict the saliency scores of image segments like superpixels [15] or object proposals [27]. For instance, Wang *et al.* [27] propose two convolutional neural networks to combine local superpixel estimation and global proposal search for salient object detection. In [15], Li *et al.* compute saliency value of each superpixel by extracting its contextual CNN features. These methods could achieve state-of-the-art saliency detection results. However, the fully connected layers added in the network decrease the computational efficiency and drop the spatial information. To handle this problem, several methods have been put forward which utilize FCN to generate a pixel-wise prediction. In [12], Lee *et al.* propose to embed low-level spatial features into the feature maps and then combine them with CNN features to predict saliency maps. Liu *et al.* [18] build a two-stage network for salient object detection. The first stage produces a coarse prediction using global structure, and the second stage hierarchically refines the details of saliency maps via integrating local context information. Wang *et al.* [29] generate saliency prior map using low-level cues and exploit it to guide the saliency prediction in a recurrent fashion. The above-mentioned methods mainly use specific-level features for generating saliency maps. Different from them, we propose a gated bi-directional message passing module to integrate multi-level features for accurately detecting salient objects. Besides, many previous works obtain the multi-scale features by feeding parallel networks with multi-context superpixels [15] or rescaled images [14]. In contrast to them, we capture rich context information for multi-level features in one network using multi-scale context-aware feature extraction module.

### 2.2. Multi-level Feature Integration

Recently, several works for dense prediction tasks [19, 6, 31] have proved that features from multiple layers are beneficial to generate better results. Features in deeper

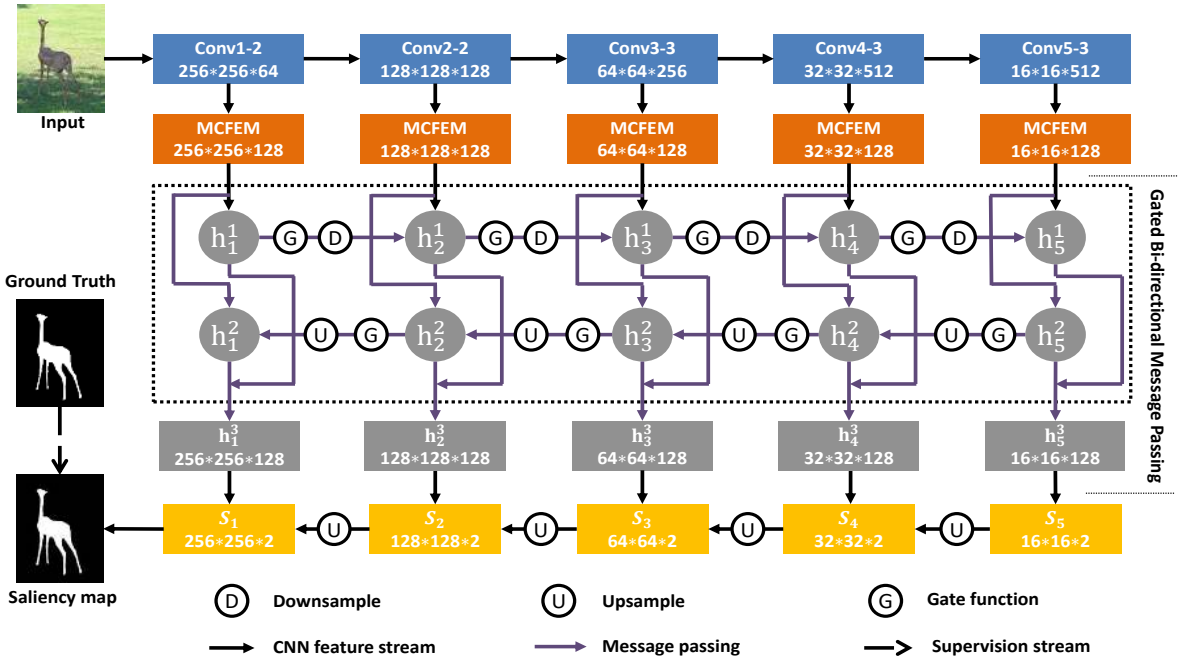



Figure 2. The overall framework of our proposed model. Each colorful box represents a feature block. Our model takes a RGB image ($256 \times 256 \times 3$) as input, and exploits VGG-16 [26] to extract multi-level features (blue boxes). Then the MCFEM (orange boxes) is used to capture context information for multi-level features. Their integration is performed by a gated bi-directional message passing module, where the gate function is employed to control the message passing rate. The integrated features $\{\mathbf{h}_i^3\}$ (gray boxes) are used for saliency prediction and the final saliency map is the fused output of multiple predicted saliency maps.

layers encode the semantic knowledge for recognizing the object categories, while features in shallower layers retain finer spatial details for reconstructing the object boundary. Some attempts [37, 8, 14] have been conducted to exploit multi-level CNN features for salient object detection. Hou *et al.* [8] propose to integrate multi-level features by building short connections from deep layers to shallow layers. In [37], Zhang *et al.* propose a feature aggregating framework, in which the multi-level CNN features are integrated into different resolutions to predict saliency maps. Li *et al.* [14] put forward a top-down refinement network, where features from deeper layers are passed to the shallower ones. Our method is clearly different from the above approaches on two aspects. Firstly, in these methods, feature integration is performed by passing features from deeper layers to the shallower ones. While our method proposes a bi-directional message passing module, where semantic information in deeper layers and fine details in shallower layers can transmit mutually. So the multiple saliency cues are incorporated at each level. Secondly, these methods always use sum or concatenation operations to integrate multi-level features without weighting their importance. While we exploit a gate function to control the message passing between different layers. The gate function could determine the on-and-off of multi-level features and selectively pass useful information from current layer to other layers.

## 3. Proposed Algorithm

In this section, we first describe the overall architecture of our proposed model in Sec. 3.1. Then we give the detailed formulas of multi-scale context-aware feature extraction module in Sec. 3.2. Sec. 3.3 provides the implementation of the gated bi-directional message passing module. At last, we introduce the saliency inference module in Sec. 3.4.

### 3.1. Overview of Network Architecture

In this paper, we propose a bi-directional message passing model to address salient object detection. Our network consists of three components: multi-scale context-aware feature extraction module, gated bi-directional message passing module and saliency inference module. The overall architecture is shown in Fig. 2. Our model is built on the FCN architecture with VGG-16 net [26] as a pre-trained model. We first feed the input image into the VGG-16 net to produce multi-level feature maps which capture different information about the salient objects. We make two modifications to the VGG-16 net, so that it can fit saliency detection task. Firstly, we discard all the fully-connected layers of the VGG-16 net as our task focuses on pixel-wise prediction. Secondly, we remove the last pooling layer to retain details of last convolutional layer. The revised VGG-16 net provides feature maps at five stages. Then we propose to add a Multi-scale Context-aware Fea-

| Layer | filter | dilation rate | output |
|---|---|---|---|
| Conv1-2 | 3*3,32 | 1/3/5/7 | 256*256*128 |
| Conv2-2 | 3*3,32 | 1/3/5/7 | 128*128*128 |
| Conv3-3 | 3*3,32 | 1/3/5/7 | 64*64*128 |
| Conv4-3 | 3*3,32 | 1/3/5/7 | 32*32*128 |
| Conv5-3 | 3*3,32 | 1/3/5/7 | 16*16*128 |

Table 1. Details of multi-scale context-aware feature extraction module (MCFEM). The "filter" with (k*k,c) means the kernel size and channel are k and c. The MCFEM first takes the feature from a side output of VGG-16 net as input. The four convolutional layers with different dilation rates are adopted to capture multi-scale context information. Finally, the cross-channel concatenation is used to integrate the multiple contextual features.

ture Extraction Module (MCFEM) after each side output of the VGG-16 net. The MCFEM is made up of convolutional layers with various fields of view and could learn multi-scale context information for different levels. To integrate the multi-level contextual features, we propose to exploit a Gated Bi-directional Message Passing Module (GBMPM). In GBMPM, semantic information in high-level features and spatial details in low-level features pass to each other in a bi-directional structure. And the gate function is used to control whether the message at the current level should be passed to the next level. We generate saliency maps at different resolutions by using these integrated multi-level features. And the final saliency map is obtained via fusing the multi-scale predicted results in a coarse-to-fine manner. Our network could be learned in an end-to-end way.

### 3.2. Multi-scale Context-aware Feature Extraction

Visual context is quite important to assist salient object detection. Existing CNN models [26, 7] learn features of objects by stacking multiple convolutional and pooling layers. However, the salient objects have large variations in scale, shape and position. Directly using the bottom-to-up single scale convolution and pooling may not effectively handle these complicated variations. A recent work [30] proposes to use the pyramid pooling before the final prediction layer to extract multi-scale features for saliency detection. However, the large scale of pooling would cause the loss of important information. Li *et al.* [13] embed the multi-scale contextual information by stacking sequential blocks containing several dilated convolutions and use the features from the last block for prediction. Inspired by this work, we propose a Multi-scale Context-aware Feature Extraction Module (MCFEM), which contains multiple dilated convolutions, to learn the information of objects and image context.

We show the details of the proposed MCFEM in Table 1. For the input image $\mathbf{I}$ with size $W \times H$, we first use the VGG-16 net to extract feature maps at five levels, which are represented as $\mathbf{F} = \{\mathbf{f}_i, i = 1, ..., 5\}$ with resolution $\tau = [\frac{W}{2^{i-1}}, \frac{H}{2^{i-1}}]$. For feature map $\mathbf{f}_i$, we then use four convolutional layers with various fields of view to capture the knowledge of object as well as image context at multiple scales. In this paper, we exploit four dilated convolutional layers [35], which could enlarge the fields of view without the loss of resolution and increasing the amount of computation, to form our MCFEM. The four dilated convolutional layers have the same convolutional kernel size $3*3$ with different dilation rates, which are set to $1, 3, 5$ and $7$ to capture multi-scale context information. We combine the feature maps from different dilated convolutional layers by cross-channel concatenation. We then apply MCFEM to multi-level feature maps, and obtain multi-scale contextual features $\mathbf{F}^c = \{\mathbf{f}_i^c, i = 1, ..., 5\}$.

In this paper, we use the dilated convolutional layers with four dilation rates (*i.e.,* 1, 3, 5 and 7) instead of convolutional filters with kernel size $3 * 3$, $7 * 7$, $11 * 11$ and $15*15$. Compared with the classic convolutional layers, the dilated convolutional layers could reduce redundant computation as well as remain the same field of view. Moreover, the experimental results in Table 3 demonstrate the advantages of our MCFEM by using dilated convolutional layers. In summary, with the MCFEM, multi-level features can encode richer context information. To integrate them for more accurate saliency prediction, we propose a Gated Bi-directional Message Passing Module.

### 3.3. Gated Bi-directional Message Passing

By adopting MCFEM, multi-level features $\mathbf{F}^c = \{\mathbf{f}_i^c, i = 1, ..., 5\}$ can capture effective context information. Besides, the semantic information at deeper layers helping localize the salient objects and spatial details at shallower ones are both important for saliency detection. In order to effectively integrate the multi-level features, we introduce a Gated Bi-directional Message Passing Module (GBMPM). Our GBMPM is inspired by the work [36] for object detection, which proposes a bi-directional model for passing messages among contextual regions of the bounding box. Different from the bi-directional model [36], which is applied at a specific layer of the backbone network, our GBMPM is built among multiple side outputs of the VGG-16. With this structure, deeper layers pass semantic information to help the shallower ones better locate the salient regions, and shallower layers transmit more spatial details to deeper ones. So multi-level features could cooperate with each other to generate more accurate results. Besides, considering that the multi-level features have various resolutions, we add upsampling and downsampling operations during the process of the bi-directional message passing.

The architecture of our bi-directional message passing module is shown in Fig. 2. It takes feature maps $\mathbf{F}^c = \{\mathbf{f}_i^c, i = 1, ..., 5\}$ with different spatial resolutions as input and outputs feature maps $\mathbf{H}^3 = \{\mathbf{h}_i^3, i = 1, ..., 5\}$. Our message passing module contains two directional connections

among multi-level features. One connection starts from the feature at the first side output layer (*i.e.*, $\mathbf{f}_1^c$) with the largest resolution ($[W, H]$) and ends at feature of the last side output layer (*i.e.*, $\mathbf{f}_5^c$) with the smallest resolution ($[\frac{W}{2^4}, \frac{H}{2^4}]$). The other direction is the opposite. Taking $\mathbf{h}_i^0 = \mathbf{f}_i^c$ with resolution $\tau = [\frac{W}{2^{i-1}}, \frac{H}{2^{i-1}}]$ for example, the process of the message passing from shallow side output to deep side output is performed by:

$$\mathbf{h}_i^1 = Down(\phi(Conv(\mathbf{h}_{i-1}^1; \theta_{i-1,i}^1))) + \phi(Conv(\mathbf{h}_i^0; \theta_i^1)) \quad (1)$$

where $Conv(*; \theta)$ is a convolutional layer with parameter $\theta = \{\mathbf{W}, \mathbf{b}\}$. $Down()$ is a shrink operation which aims to downsample the feature map by a factor of 2 to adapt the size of higher level feature map. And $\phi()$ is a ReLU activation function. $\mathbf{h}_i^1$ is the updated features after receiving message from lower-level feature $\mathbf{h}_{i-1}^1$. Note that we set $\mathbf{h}_0^1 = 0$, since $\mathbf{h}_1^1$ is from the first side output and receives no message from the former layers. The opposite direction of message passing from deep layer to shallow layer is:

$$\mathbf{h}_i^2 = Up(\phi(Conv(\mathbf{h}_{i+1}^2; \theta_{i,i+1}^2))) + \phi(Conv(\mathbf{h}_i^0; \theta_i^2)) \quad (2)$$

where $Up()$ is an operation to upsample the feature map by a factor of 2. And $\mathbf{h}_i^2$ represents the updated features after receiving message from $\mathbf{h}_{i+1}^2$ with high-level information. We also take $\mathbf{h}_6^2 = 0$, as $\mathbf{h}_5^2$ is from the last side output layer and receives no message from latter layers. After the bi-directional message transmission using Eq. 1 and Eq. 2, the features in $\mathbf{h}_i^1$ could receive more spatial details from low-level features and features in $\mathbf{h}_i^2$ receive semantic messages from high-level features. In order to form a better representation for the *i-th* side output layer, we incorporate the features from both directions as follows:

$$\mathbf{h}_i^3 = \phi(Conv(Cat(\mathbf{h}_i^1, \mathbf{h}_i^2); \theta_i^3)) \quad (3)$$

where $Cat()$ is the concatenation operation among channel axis. With Eq. 3, $\mathbf{h}_i^3$ contains both high-level semantic and low-level spatial information. The multi-level features $\{\mathbf{h}_i^3\}_{i=1,...,5}$ are robust and will be jointly used for salient object prediction.

For the individual input image, the multi-level features may be not all helpful to precisely predict saliency maps. Instead of passing messages without selection for all the input images, we exploit gate function [36] to make the message adaptively pass among multi-level features. Our motivation is that during the process of message passing, a decision should be made about whether the message of the current level is useful for feature of the next level. The gate function is designed as convolutional layers with sigmoid activation to produce message passing rate in the range of $[0, 1]$, which is used to control the message passing. With the gate function, the message passing of bi-directional structure in Eq. 1 and Eq. 2 are changed as:

$$\begin{aligned}\mathbf{h}_i^1 =& Down(G(\mathbf{h}_{i-1}^0; \theta_{i-1,i}^{g1}) \otimes \phi(Conv(\mathbf{h}_{i-1}^1; \theta_{i-1,i}^1))) \\ &+ \phi(Conv(\mathbf{h}_i^0; \theta_i^1)) \quad (4)\\ \mathbf{h}_i^2 =& Up(G(\mathbf{h}_{i+1}^0; \theta_{i,i+1}^{g2}) \otimes \phi(Conv(\mathbf{h}_{i+1}^2; \theta_{i,i+1}^2))) \\ &+ \phi(Conv(\mathbf{h}_i^0; \theta_i^2)) \quad (5)\end{aligned}$$

where $\otimes$ is element-wise product. $G(*; \theta^g)$ is the gate function to control the message passing rate, which is defined as

$$G(\mathbf{x}; \theta^g) = Sigm(Conv(\mathbf{x}; \theta^g)) \quad (6)$$

where $Sigm()$ is the element-wise sigmoid function. $Conv(\mathbf{x}; \theta^g)$ is a $3 * 3$ convolutional layer having the same number of channels with $\mathbf{x}$, which means the gate function learns a different gated filter for each channel of $\mathbf{x}$. When $G(\mathbf{x}; \theta^g) = 0$, the message of $\mathbf{x}$ is blocked and would not be passed to other levels. And the formulation for generating $\mathbf{h}_i^3$ is unchanged. By adding the gate function into the bi-directional message passing module, only useful information is passed between different levels, and the inaccurate information is prevented. With the GBMPM, multi-level features $\mathbf{h}_i^3$ can adaptively encode various saliency cues and are robust enough to produce accurate saliency prediction.

### 3.4. Saliency Inference

In the above sections, we use the MCFEM to capture multi-scale context information for each side output of the VGG-16 net. And the multi-level features are further processed via GBMPM, so they simultaneously contain semantic information and fine details. The multi-level features are complementary and robust, so we use them together to predict saliency maps. Some methods [37, 30] directly upsample and integrate the multi-level features into the size of the input image and exploit a convolutional layer to produce saliency map. However, this upsampling operation may lead to the loss of details of the detected objects. To avoid this problem, we fuse multi-level features and generate saliency map in a coarse-to-fine manner. Our fusion module takes the feature map $\mathbf{h}_i^3$ (resolution is $[\frac{W}{2^{i-1}}, \frac{H}{2^{i-1}}]$) and the high-level prediction $\mathbf{S}_{i+1}$ as input. The fusion process is summarized as follows:

$$\mathbf{S}_i = \begin{cases} Conv(\mathbf{h}_i^3; \theta_i^f) + Up(\mathbf{S}_{i+1}), i < 5 \\ Conv(\mathbf{h}_i^3; \theta_i^f), i = 5 \end{cases} \quad (7)$$

where $Conv(*; \theta^f)$ is the convolutional layer with kernel size $1 * 1$ for predicting saliency maps. Using Eq. 7, predictions from deep layers are hierarchically and progressively transmitted to shallow layers. And we take $\mathbf{S}_1$ as the final saliency map of our model without any post-processing. The proposed model is trained end to end using the cross-entropy loss between the final saliency map and the ground truth. The loss function is defined as:

$$L = -\sum_{x,y} l_{x,y} log(P_{x,y}) + (1 - l_{x,y}) log(1 - P_{x,y}) \quad (8)$$

$l_{x,y} \in \{0, 1\}$ is the label of the pixel $(x, y)$, and $P_{x,y}$ is the probability of pixel $(x, y)$ belonging to the foreground.

# 4. Experiments

## 4.1. Experimental Setup

**Dataset.** We evaluate the proposed method on five benchmark datasets: ECSSD [32], PASCAL-S [17], SOD [23], HKU-IS [15] and DUTS [28]. The ECSSD dataset [32] has 1000 images with various complex scenes. The PASCAL-S dataset [17] is based on the validation set of the PASCAL VOC 2009 segmentation challenge. Images in this dataset are much more complicated with cluttered backgrounds and multiple objects. This dataset contains 850 natural images with pixel-wise annotations. The SOD dataset has 300 images selected from the Berkeley segmentation dataset [22]. It is one of the most difficult saliency datasets currently. The HKU-IS dataset proposed in [15] has 4447 images and most of the images include multiple disconnected salient objects. DUTS [28] is a large scale dataset, which contains 10553 images for training and 5019 images for testing. The images are challenging with salient objects of various locations and scales as well as complex background. We use this dataset for both training and testing our proposed model.

**Evaluation Criteria.** We evaluate the performance of the proposed model as well as other state-of-the-art salient object detection methods using three metrics, including precision-recall (PR) curves, F-measure and mean absolute error (MAE). The precision value is the ratio of ground truth salient pixels in the predicted salient region. And the recall value is defined as the percentage of the detected salient pixels and all ground truth area. The precision and recall are calculated by thresholding the predicted saliency map and comparing it with the corresponding ground truth. Taking the average of precision and recall of all images in the dataset, we can plot the precision-recall curve at different thresholds. The F-measure is an overall performance indicator, it is computed by the weighted harmonic of precision and recall:

$$F_\beta = \frac{(1 + \beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recall} \quad (9)$$

where $\beta^2$ is set to 0.3 to weight precision more than recall as suggested in [33]. The F-measure curve is obtained by connecting F-measure scores under different thresholds. We report the maximum F-measure from all precision-recall pairs, which is a good summary of the method's detection performance [2]. Except for PR curve and F-measure, we also calculate the mean absolute error (MAE) to measure the average difference between predicted saliency map and ground truth. It is computed as:

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

where $S$ and $G$ are predicted saliency map and ground truth, respectively.

**Implementation Details.** We utilize the training set of DUTS dataset [28] to train our proposed model. It contains 10553 images with high-quality pixel-wise annotations. We augment the training set by horizontal flipping and cropping the images to relieve the over-fitting problem, as suggested in [18]. We don't use the validation set and train the model until its training loss converges. A NVIDIA Titan X GPU is used for training and testing. The parameters of the first 13 convolutional layers are initialized by VGG-16 net [26]. For other convolutional layers, we initialize the weights using truncated normal method. The convolutional parameters of our message passing module in Sec. 3.3 are not shared, and the upsampling and downsampling are conducted simply by bilinear interpolation. Our model is trained using Adam [11] with an initial learning rate at 1e-6. The training process of our model takes about 30 hours and converges after 12 epochs. During testing, our proposed model runs about 22 fps with $256 \times 256$ resolution.

## 4.2. Performance Comparison with State-of-the-art

We compare the proposed saliency detection model with 13 state-of-the-art methods, including 11 deep learning based methods (LEGS [27], MDF [15], RFCN [29], ELD [12], DCL [16], DHS [18], NLDF [20], DSS [8], Amulet [37], UCF [38], SRM [30]) and 2 conventional ones (DRFI [9] and BSCA [24]). For fair comparison, the saliency maps of different methods are provided by the authors or achieved by running available codes or softwares.

**Quantitative Evaluation.** We perform comparisons of the proposed algorithm and 13 state-of-the-art saliency detection methods on five datasets. The comparison results are shown in Fig 3 and Table 2. Table 2 illustrates the performances of different methods under the metrics of maximum F-measure and MAE. Our method can consistently outperform other approaches across all the datasets in terms of different measurements, which demonstrates the effectiveness of the proposed model. For F-measure and MAE, our method achieves the best results among five datasets. Note that the MDF [15] use 3000 images from HKU-IS [15] dataset for training its model, so we don't provide its comparison result on this dataset.

Fig. 3 lists the PR curves and F-measure curves of different approaches on five datasets. We can observe that the PR curves of the proposed algorithm perform better than other methods on five datasets. Besides, the F-measure curves of our method are significantly higher than other methods, which means our method is more robust than other approaches even on challenging datasets.

| * | ECSSD | | PASCAL-S | | SOD | | HKU-IS | | DUTS-test | |
|---|---|---|---|---|---|---|---|---|---|---|
| | max $F_\beta$ | MAE | max $F_\beta$ | MAE | max $F_\beta$ | MAE | max $F_\beta$ | MAE | max $F_\beta$ | MAE |
| Ours | 0.928 | 0.044 | 0.862 | 0.074 | 0.851 | 0.106 | 0.920 | 0.038 | 0.850 | 0.049 |
| SRM [30] | 0.917 | 0.054 | 0.847 | 0.085 | 0.839 | 0.126 | 0.906 | 0.046 | 0.827 | 0.059 |
| DSS [8] | 0.916 | 0.052 | 0.836 | 0.096 | 0.841 | 0.118 | 0.910 | 0.041 | 0.825 | 0.057 |
| Amulet [37] | 0.915 | 0.059 | 0.837 | 0.098 | 0.802 | 0.141 | 0.895 | 0.052 | 0.778 | 0.085 |
| UCF [38] | 0.911 | 0.078 | 0.828 | 0.126 | 0.798 | 0.164 | 0.886 | 0.074 | 0.771 | 0.117 |
| NLDF [20] | 0.905 | 0.063 | 0.831 | 0.099 | 0.837 | 0.123 | 0.902 | 0.048 | 0.812 | 0.066 |
| DHS [18] | 0.907 | 0.059 | 0.829 | 0.094 | 0.822 | 0.127 | 0.890 | 0.053 | 0.807 | 0.067 |
| DCL [16] | 0.890 | 0.088 | 0.805 | 0.125 | 0.820 | 0.139 | 0.885 | 0.072 | 0.782 | 0.088 |
| ELD [12] | 0.867 | 0.079 | 0.773 | 0.123 | 0.760 | 0.154 | 0.839 | 0.074 | 0.738 | 0.093 |
| RFCN [29] | 0.890 | 0.107 | 0.837 | 0.118 | 0.802 | 0.161 | 0.892 | 0.079 | 0.784 | 0.091 |
| LEGS [27] | 0.827 | 0.118 | 0.762 | 0.155 | 0.729 | 0.195 | 0.766 | 0.119 | 0.655 | 0.138 |
| MDF [15] | 0.832 | 0.105 | 0.768 | 0.146 | 0.783 | 0.155 | - | - | 0.730 | 0.094 |
| DRFI [9] | 0.786 | 0.164 | 0.698 | 0.207 | 0.697 | 0.223 | 0.777 | 0.145 | 0.647 | 0.175 |
| BSCA [24] | 0.758 | 0.182 | 0.667 | 0.223 | 0.653 | 0.251 | 0.719 | 0.175 | 0.597 | 0.197 |

Table 2. The maximum F-measure (larger is better) and MAE (smaller is better) of different saliency detection methods on five released saliency detection datasets. The best three results are shown in red, green and blue.

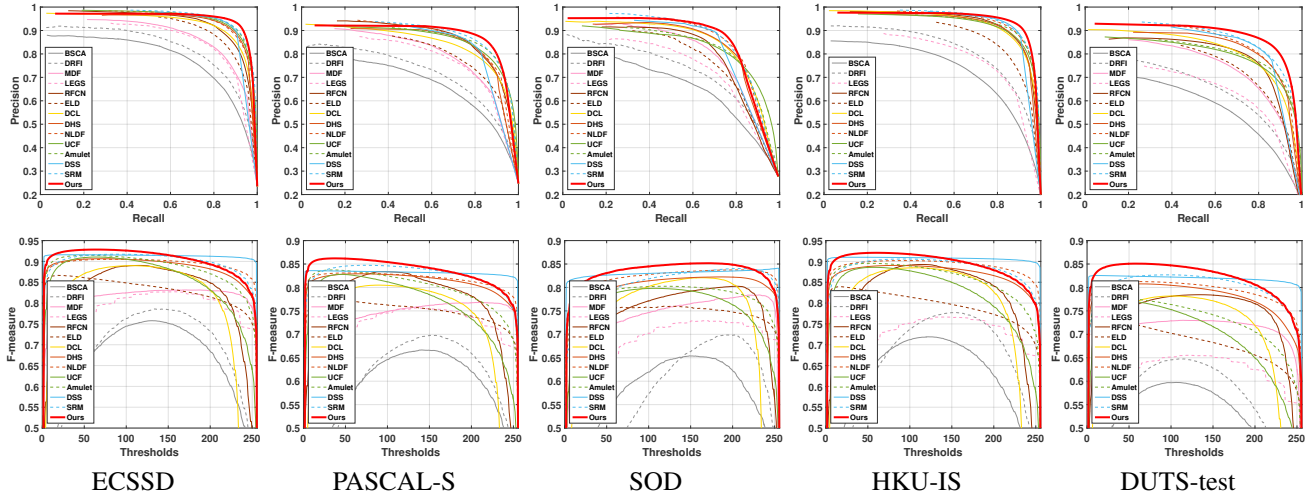



Figure 3. Quantitative comparisons of the proposed approach and 13 baseline methods on five datasets. The first and second rows are the PR curves and F-measure curves of different methods, respectively.

**Qualitative Evaluation.** Fig. 4 lists some saliency maps generated by the proposed method as well as other 13 state-of-the-art algorithms. The images are selected from five datasets for testing. It can be seen that our method can accurately detect salient objects. For objects with various scales and shapes, our method can highlight the entire objects with well-defined boundaries (the 1-3 rows). Our method is also robust for images with multiple objects (4-5 rows) and complex background (6-8 rows).

### 4.3. Analysis of the Proposed Approach

The proposed framework is composed of two modules, including the multi-scale context-aware feature extraction (MCFEM) and the gated bi-directional message passing (GBMPM). To investigate the effectiveness of each module, we conduct a series of experiments on ECSSD [32] dataset.

**The Effectiveness of MCFEM.** We propose the MCFEM to capture more context information to detect objects with various scales. To highlight the advantages of the MCFEM, we provide another three methods for comparisons. The first one (named as "FCN") is to employ the multiple side output features of the VGG-16 net to predict saliency maps by using the inference method in Sec. 3.4. And the second one (named as "MCFEM with convolutional layer") is that we replace the dilated convolutional layers with dilation rates $r = 1, 3, 5, 7$ by convolutional layers with kernel size $k = 3, 7, 11, 15$. The last one (named as "pyramid pooling") is a pyramid pooling module similar to PSPNet [39], in which the pooling kernels are set to 3, 7, 11, 15, respectively. Table 3 shows the maximum F-measure and MAE of the above-mentioned models on ECSSD dataset. We can observe that the proposed MCFEM is effective in salient object detection, especially implemented with dilated convolutional layers, which outperforms the

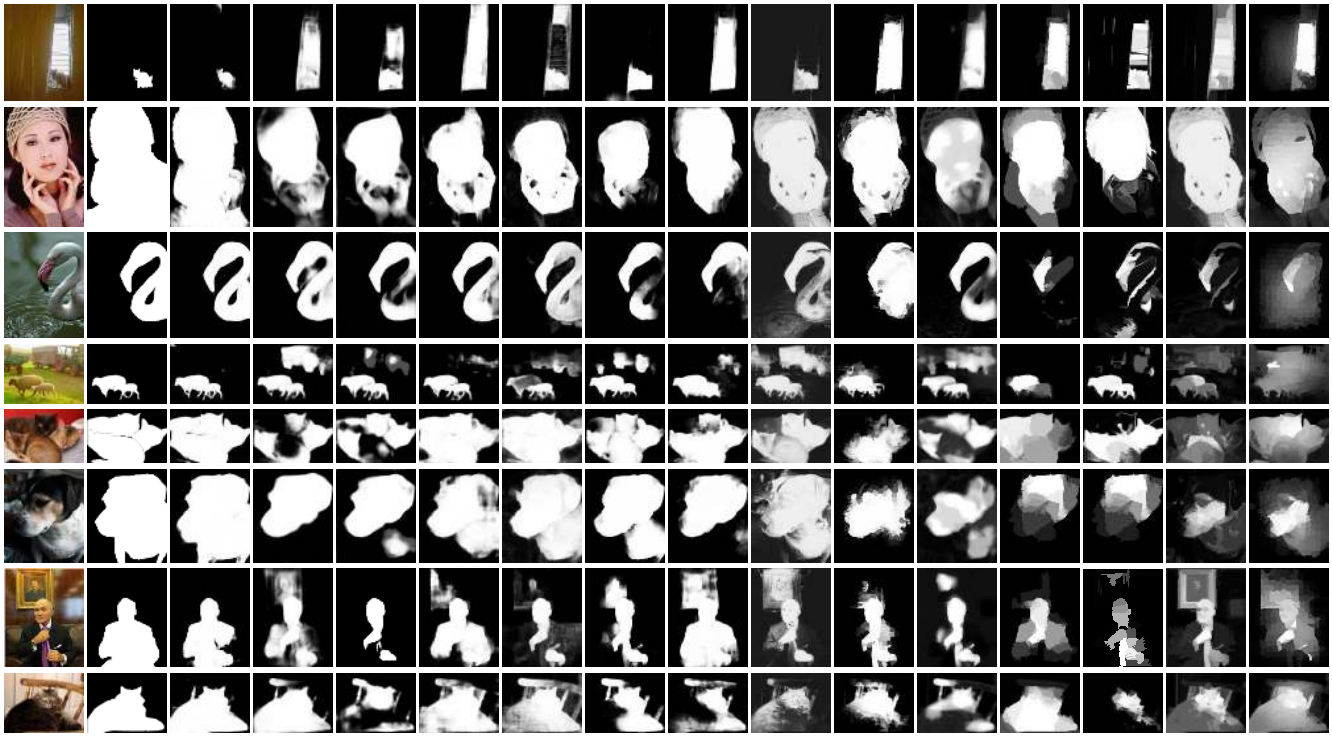



Figure 4. Qualitative comparisons of the proposed method and the state-of-the-art algorithms.

| model setting | max $F_\beta$ | MAE |
|---|---|---|
| FCN (baseline) | 0.887 | 0.068 |
| pyramid pooling | 0.894 | 0.061 |
| MCFEM with convolutional layer | 0.909 | 0.058 |
| MCFEM with dilated convolutional layer | 0.914 | 0.053 |
| Downsampling stream in GBMPM | 0.894 | 0.063 |
| Upsampling stream in GBMPM | 0.903 | 0.060 |
| GBMPM with closed gate | 0.889 | 0.068 |
| GBMPM with open gate | 0.908 | 0.059 |
| GBMPM | 0.917 | 0.052 |
| MCFEM + GBMPM | **0.928** | **0.044** |

Table 3. Quantitative comparisons of different settings in the MCFEM and GBMPM.

methods of FCN, piramid pooling and the MCFEM with convolutional layers, respectively.

**The Effectiveness of GBMPM.** In Sec.3.3, we propose a gated bi-directional message passing module (GBMPM) to adaptively incorporate multi-level features. The GBMPM consists of two components, the bi-directional message passing structure and the gate function. To prove their contributions to the model, we implement various settings in the GBMPM, and their maximum F-measure and MAE are shown in Table 3. We remove the MCFEM to better investigate the performance of the components in GBMPM. We obtain the results of message passing from deep side outputs of FCN to shallow ones ( named as "Upsampling stream in GBMPM") and message passing in the opposite direction ("Downsampling stream in GBMPM"), respectively. The comparison with our bi-directional message passing model ("GBMPM") demonstrates the advantage of the bi-directional structure in GBMPM. We also verify the contribution of the gate function in GBMPM. The results when all gates are open ("GBMPM with open gate") or closed ("GBMPM with closed gate") are shown in Table 3. Compared with "GBMPM", the proposed adaptive gate function is effective. Besides, the comparison between "MCFEM with dilated convolutional layer", "GBMPM" and "MCFEM+GBMPM" verifies that both MCFEM and GBMPM contribute to the final result.

## 5. Conclusion

In this paper, we propose a novel bi-directional message passing model for salient object detection. We first design a multi-scale context-aware feature extraction module, which consists of dilated convolutions with multiple fields of view and captures objects and image context at multiple scales. Then we introduce a gated bi-directional message passing module to integrate multi-level features, in which features from different levels adaptively pass messages to each other. The multi-level features containing both high-level semantic concept and low-level spatial details are further utilized to produce the final saliency maps. Experimental results on five datasets demonstrate that our proposed approach outperforms 13 state-of-the-art methods under different evaluation metrics.

**Acknowledgements.** This work was supported by the Natural Science Foundation of China under Grant 61725202 and 61472060.

## References


[1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[2] A. Borji, M. M. Cheng, H. Jiang, and J. Li. Salient object detection: A benchmark. *IEEE Transactions on Image Processing*, 24(12):5706–5722, 2015.

[3] A. Borji, S. Frintrop, D. N. Sihite, and L. Itti. Adaptive object tracking by learning background context. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[4] M. M. Cheng, G. X. Zhang, N. J. Mitra, and X. Huang. Global contrast based salient region detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[5] Y. Gao, M. Wang, D. Tao, R. Ji, and Q. Dai. 3-d object retrieval and recognition with hypergraph analysis. *IEEE Transactions on Image Processing*, 21(9):4290–4303, 2012.

[6] B. Hariharan, P. Arbelez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[8] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. Torr. Deeply supervised salient object detection with short connections. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[9] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li. Salient object detection: A discriminative regional feature integration approach. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[10] Z. Jiang and L. S. Davis. Submodular salient region detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[11] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[12] G. Lee, Y. Tai, and J. Kim. Deep saliency with encoded low level distance map and high level features. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[13] D. Li, X. Chen, Z. Zhang, and K. Huang. Learning deep context-aware features over body and latent parts for person re-identification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[14] G. Li, Y. Xie, L. Lin, and Y. Yu. Instance-level salient object segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[15] G. Li and Y. Yu. Visual saliency based on multiscale deep features. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[16] G. Li and Y. Yu. Deep contrast learning for salient object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[17] Y. Li, X. Hou, C. Koch, J. Rehg, and A. Yuille. The secrets of salient object segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[18] N. Liu and J. Han. Dhsnet: Deep hierarchical saliency network for salient object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[19] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[20] Z. Luo, A. Mishra, A. Achkar, J. Eichel, S. Li, , and P. M. Jodoin. Non-local deep features for salient object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[21] V. Mahadevan and N. Vasconcelos. Saliency-based discriminant tracking. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[22] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of the IEEE International Conference on Computer Vision*, 2002.

[23] V. Movahedi and J. H. Elder. Design and perceptual validation of performance measures for salient object segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[24] Y. Qin, H. Lu, Y. Xu, and H. Wang. Saliency detection via cellular automata. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[25] Z. Ren, S. Gao, L. T. Chia, and W. H. Tsang. Region-based saliency detection and its application in object recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 24(5):769–779, 2014.

[26] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[27] L. Wang, H. Lu, X. Ruan, and M.-H. Yang. Deep networks for saliency detection via local estimation and global search. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[28] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan. Learning to detect salient objects with image-level supervision. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[29] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan. Saliency detection with recurrent fully convolutional networks. In *Proceedings of European Conference on Computer Vision*, 2016.

[30] T. Wang, A. Borji, L. Zhang, P. Zhang, and H. Lu. A stagewise refinement model for detecting salient objects in images. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[31] S. Xie and Z. Tu. Holistically-nested edge detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[32] Q. Yan, L. Xu, J. Shi, and J. Jia. Hierarchical saliency detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[33] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang. Saliency detection via graph-based manifold ranking. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[34] W. Yang, W. Ouyang, H. Li, and X. Wang. End-to-end learning of deformable mixture of parts and deep convolutional neural networks for human pose estimation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[35] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[36] X. Zeng, W. Ouyang, B. Yang, J. Yan, and X. Wang. Gated bi-directional cnn for object detection. In *Proceedings of European Conference on Computer Vision*, 2016.

[37] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan. Amulet: Aggregating multi-level convolutional features for salient object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[38] P. Zhang, D. Wang, H. Lu, H. Wang, and B. Yin. Learning uncertain convolutional features for accurate saliency detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[39] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[40] R. Zhao, W. Ouyang, and X. Wang. Unsupervised salience learning for person re-identification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[41] W. Zhu, S. Liang, Y. Wei, and J. Sun. Saliency optimization from robust background detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2014.