MULTI-SCALE CAPSULE ATTENTION BASED SALIENT OBJECT DETECTION WITH MULTI-CROSSED LAYER CONNECTIONS

Anonymous ICME submission

ABSTRACT

With the popularization of convolutional networks being used for saliency models, saliency detection performance has achieved significant improvement. However, how to integrate accurate and crucial features for modeling saliency is still underexplored. In this paper, we present **CapSalNet**, a multi-scale Capsule attention module based Salient object detection deep Network. We first propose a novel capsule attention to integrate multi-scale contextual information with dynamic routing. Then, our model adaptively learns to aggregate multi-level features by using multi-crossed skip-layer connections. Finally, the predicted results are efficiently fused to generate the final saliency map in a coarse-to-fine manner. Comprehensive experiments on four benchmark datasets demonstrate that our proposed algorithm outperforms existing state-of-the-art approaches.

Index Terms— Capsule attention, multi-crossed layer connections, salient object detection

1. INTRODUCTION

Salient object detection aims to localize the most conspicuous and eye-catching objects or regions in an image. It can be taken as a pre-processing step in many computer vision tasks, such as scene classification [1], visual tracking [2] and person re-identification [3].

In the early stages, inspired by human visual attention mechanisms, many conventional salient object detection models are proposed, mainly based on various low-level handcrafted features for detecting salient objects, including global contrast [4] and background priors [5]. With the advance of deep learning, many Convolutional Neural Network (CNN)-based saliency models have been proposed. CNN-based multi-level features are introduced. Most existing deep saliency models focus on how to better aggregate various features. Early models combine hand-crafted features with high-level features [6] and integrate patch-level features with pixel-level feautures [7] to complement enough information. But those patch-level saliency maps and hand-crafted features are typically blurry and time-consuming. Recent models pay more attention to pixel-level convolutional feature aggregation. End-to-end deep Fully Convolutional Networks (FCNs) for salient object detection [8, 9, 10, 11, 12, 13] aggregate different feature maps in different layers. Those models show the excellent performance with the supervision of large scale of datasets. Besides, Wang et al. [14] develops a new saliency model using recurrent fully convolutional networks (RFCNs) to automatically learn to refine the saliency map by correcting its previous errors.

However, existing FCN-based networks directly aggregate multi-level features extracted from the entire raw image, which introduces redundant background information and misleads the networks to focus on non-salient regions. To cope with the problem, in this paper, we propose a novel multi-scale Capsule attention based Salient object detection deep Network, namely **CapSalNet**, which aggregates the features selectively rather than indiscriminately. Capsule is first proposed by Sabour et al. [15], as a vector of neurons, whose orientation represents the properties of the entity and whose length represents the existence of the entity. The routing-by-agreement algorithm measures degree of agreement through a scalar product between two capsule vectors. We introduce capsules into a convolutional network as the attention module to learn the weight of multi-level features. To the best of our knowledge, there is no previous work that introduces the capsule into salient object detection.

Moreover, most of the state-of-the-art CNN models estimate saliency from multi-scale side-output convolutional features by using kernels of different sizes. In our algorithm, multi-scale features are extracted with different dilated rates, but involved the same kernel size. This can reduce the number of parameters. Then, we utilize the capsule attention module to selectively learn the multi-scale features with dynamic routing for salient object representation. After that, in order to preserve the semantic information in deep layers and spatial details in shallow layers simultaneously, we propose the multi-crossed layer connections for multi-level feature aggregation. Finally, the generation of a saliency map is inferred by means of deep supervision with fusion of multi-scale predicted results.

Our main contributions are summarized as follows:

- We propose a novel multi-scale capsule attention mechanism, which selectively integrates multi-scale contextual information with dynamic routing to make the network pay attention to salient regions.
- We propose a multi-crossed layer feature aggregation

module based on a deeply recursive supervision learning framework. This module effectively incorporates edge-aware feature maps in the low-level layers, and semantic-aware feature maps in the high-level layers, by using skip-layer connections to achieve accurate object boundary inference and semantic enhancement.

- The proposed model achieves favorable state-of-the-art performance on large-scale salient object detection datasets, including the recent DUT-OMRON [16], ECSSD [17], HKU-IS [18], PASCAL-S [19]. In addition, the model is fast on modern GPUs, which achieves a real-time speed of nearly 27 fps.

2. PROPOSED ALGORITHM

2.1. Overview of the Network Architecture

Figure 1 shows our proposed deep end-to-end network architecture for salient object detection. Our model is built on the FCN architecture, with VGG-16 net[20] as a pretrained model for fast convergence. Then, we propose a Multi-Scale Capsule Attention (MSCA) module to be added after each side-output of the VGG-16 net. The MSCA module can attentively learn multi-scale features, and is described in Section 2.2. After that, we propose to exploit a Multi-crossed Layer Connections (MLC) module to integrate the features at different levels, which will be introduced in detail in Section 2.3. The integrated features are represented as N_i^1 and N_i^2. Finally, we fuse saliency maps at different resolutions in a coarse-to-fine manner to preserve the boundary details in shallow layers and semantic cues in deep layers. S_1 is the final saliency map.

The fusion process is summarized as:

$$S_i = \begin{cases} \sigma(W_f * Cat(N_1^i, N_2^i)) + b_f) + Up(S_{i+1}), & i < 5 \\ \sigma(W_f * Cat(N_1^i, N_2^i)) + b_f), & i = 5 \end{cases} \tag{1}$$

where W_f represents the fusion parameters with kernel size 1×1 and b_f is the bias parameter. Cat represents the concatenate operation. σ is the sigmoid activation function. $Up()$ is an operator to upsample the feature map by a factor of 2. S_i is learned by the deep supervision for fast convergence. The loss function in our network is the average value of square loss between saliency map S_i and ground truth S_{gt}. It is defined as follows:

$$L = Mean(\sum_{i=0}^{5} \|S_i - S_{gt}\|^2) \tag{2}$$

where $Mean()$ is the pixel mean. The ground truth is resized to the same size as the corresponding S_i by bilinear interpolation.

2.2. Multi-scale Capsule Attention Module

Salient Objects have large variations in terms of scale and shape. To solve this problem, four dilated convolutional layers [21] are used, each with a different kind of view after the side-output of the VGG-16 net. It can capture multi-scale features for salient object detection. As shown in the right sub figure in Figure 1, F represents the feature layer from VGG-16 convolutional block. We utilize the four dilated convolutional layers have the same kernel size of 3×3, with dilation rates of 1,3,5 and 7 respectively. Then, the feature maps from the different dilated convolutional layers are concatenated as F_s. It can be calculated as Eq. 3. The dilated convolutional layers reduce redundant computation, and at the same time, obtain various fields of view.

$$F_s = Cat(\sigma(W_d^j * F + b_j)), j = 1, 2, 3, 4. \tag{3}$$

where W_d^j represents the parameter of a dilated convolutional layer.

Directly aggregating the current multi-scale features F_s can cause the problem of distraction from background regions. Therefore, we propose a novel capsule attention mechanism to learn more effective feature maps to make the network pay more attention to the foreground regions. F_c is a convolutional capsule layer with 128 channels of convolutional capsules (each capsule contains one convolutional units with a 3×3 kernel.) F_c is calculated as:

$$F_c = squash(W_c * F_s + b_c) \tag{4}$$

where W_c represents the parameters of the convolutional capsule layer and b_c is the bias term. The non-linear squash function is defined in Eq. 5, which ensures that a short vector will shrink to almost zero length and a long vector shrinks to a length slightly below 1.

$$squash(x) = \frac{\|x\|^2}{0.5 + \|x\|^2} \frac{x}{\|x\|^2} \tag{5}$$

where x represents the $W_c * F_s + b_s$ in Eq. 4. After that, the dynamic routing function, described as Algorithm 1, is operated on the capsule layer F_c to obtain capsule attention (ca). We explore the transformation matrices to generate the prediction vector $u_{n|m}$ from a child capsule m to its parent capsule n. After that, ca is applied to each slice of F_s for the attention weights as follows:

$$F_{sca} = ca \otimes F_s. \tag{6}$$

where $\otimes$ represents element multiplication.

2.3. Multi-crossed Layer Connections

After the multi-scale capsule attention module, multi-level features are selectively obtained. The deep layers contain semantic information, while the shallower layers preserve the spatial details. Therefore, the multi-level features need to be integrated to achieve accurate saliency detection.

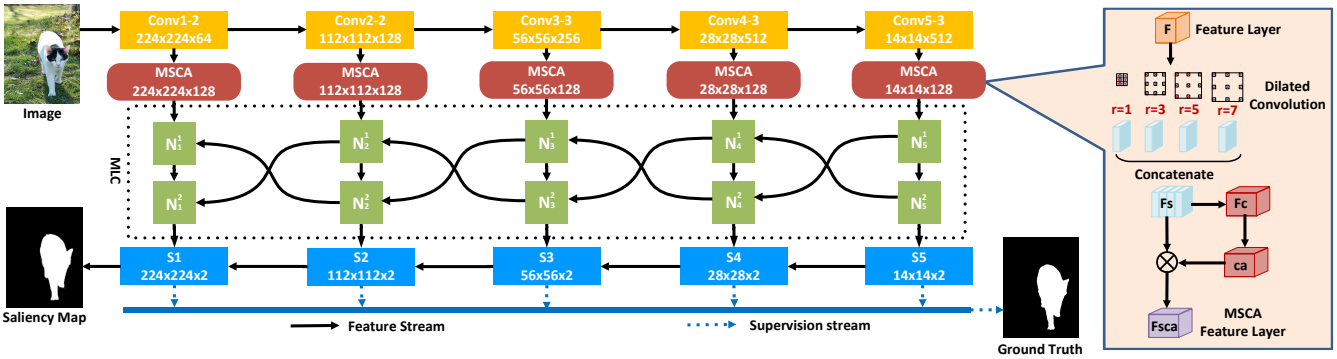


Fig. 1. The overall architecture of our proposed CapSalNet. Each colored box represents a feature block, and the input is a RGB image ($224 \times 224 \times 3$). The black arrows between blocks indicate the feature stream. Firstly, the modified VGG-16 net is exploited to extract features (yellow boxes). Secondly, the multi-scale capsule attention (MSCA) modules are used to learn the multi-scale features selectively (red boxes). Each MSCA is desrcibed in The right sub figure which is detailedly demonstrated in Sec. 2.2. Thirdly, the integrated features N_i^1 and N_i^2 (green boxes) are obtained by multi-crossed layer connections (MLC). Finally, the saliency map is generated by fusing the multi-level features in a coarse-to-fine manner. Deep supervision is employed to improve the interaction of multiple predictions, shown as blue dotted lines.

Algorithm 1 Dynamic Routing Algorithm

Require:

procedure ROUTING($\hat{u}_{n|m}, \hat{a}_{n|m}, r, l$);
Initialize the logits of coupling coefficients $b_{n|m} = 0$;
for r iterations do
for all capsule m in layer l and capsule n in layer $l+1$ do
$c_{n|m} = \hat{a}_{n|m} \cdot softmax(b_{n|m})$
for all capsule n in layer $l+1$ do
$v_n = squash(\sum_m c_{n|m} \hat{u_{n|m}}), a_n = \|v_n\|$ (5)
for all capsule m in layer l and capsule n in
$b_{n|m} = b_{n|m} + \hat{u}_{n|m} \cdot v_n$
return v_n, a_n

Multi-crossed Layer Connections (MLC) module is shown in black dotted box in Figure 1. N_i^1, and $N_i^2 (i = 1, 2, \cdots, 5)$ represent the outputs of the convolutional layer, whose kernel size is 3×3, are multi-level integrated. The shape of the connection between the layers is visualized as a cross. We abstract the connections in different layers as the multi-crossed layer connections. The addition function combines the different layer features. When $i = 1, 2, 3$, and 4, these tow outputs are computed as follows:

$$\begin{aligned} N_1^i &= \phi((W_c^i * F_s ca^i + b_c) + Up(\sigma(W_c^i * N_2^{i+1} + b_c))), \\ N_2^i &= \phi((W_c^i * Conv^i + b_c) + Up(\sigma(W_c^i * N_1^{i+1} + b_c))) \end{aligned} \tag{7}$$

When i is equal to 5, N_1^i and N_2^i are defined as:

$$\begin{aligned} N_1^i &= \phi(W_c^i * F_s ca^i + b_c) \\ N_2^i &= \phi(W_c^i * Conv^i + b_c) \end{aligned} \tag{8}$$

The $Conv$ operation in Eq. 8 is calculated as:

$$Conv^i = \phi(W_c^i * N_1^i + b_c), (i = 1, 2, 3, 4, 5) \tag{9}$$

where W_c^i represents the parameters of the convolutional layer with 3×3 kernel size. ϕ is the ReLU activation function. The use of skip-layer connection can effectively aggregate the multi-level features.

3. EXPERIMENTS

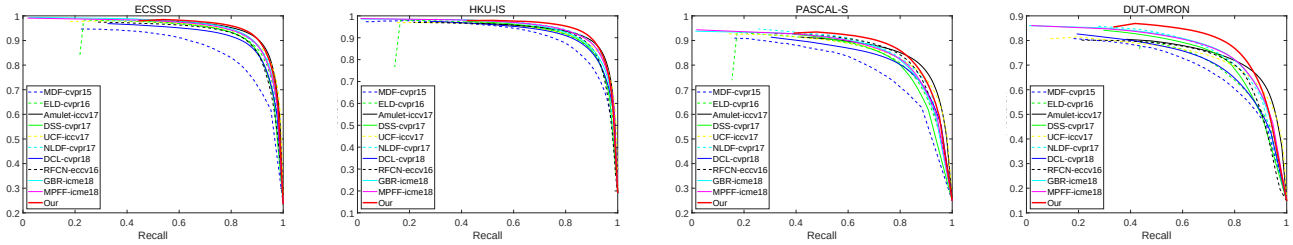
3.1. Experimental Setup

Datasets: For performance evalutation, we compares the proposed method on four benchmark datasets: ECSSD [17], PASCAL-S [19], HKU-IS [18] and DUT-OMRON [16]. ECSSD has 1,000 images with various complex scenes. PASCAL-S contains 850 natural images with pixel annotations, which are complicated with cluttered backgrounds and multiple objects. HKU-IS has 4,447 challenging images and most of the images include multiple disconnected salient objects. DUT-OMRON contains 5,168 challenging images with complex backgrounds.

Evaluation Metrics: We evaluate the performance of the proposed model, as well as other state-of-the-art salient object detection methods using three metrics, including the precision-recall (PR) curves, F-measure and mean absolute error (MAE). Precision measured as the percentage of ground-truth salient pixels in a predicted salient region, and recall is defined as the percentage of the detected salient pixels and all of the ground-truth area. F-measure is defined as follows:

$$Fmeasure = \frac{(1 + \beta^2) \cdot Precision \cdot Recall)}{\beta^2 \cdot Precision + Recall} \tag{10}$$

where β^2 is set to 0.3 to weight precision more than recall, as suggested in [16]. Futhermore, we report the maximum F-



(a) ECSSD (b) HKU-IS (c) PASCAL-S (d) DUT-OMRON

Fig. 2. The performance of PR curves on four datasets.

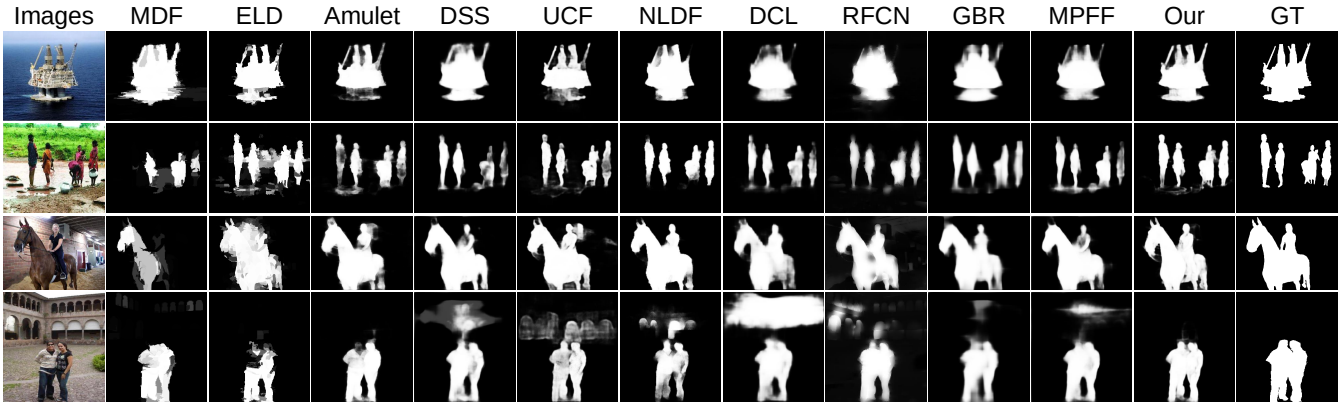


Fig. 3. Visual comparison of our results compared with state-of-the-art methods.

measure from all the precision-recall pairs, which is a good summary of the detection performance [22]. MAE measures the average pixel-level absolute difference between the detected saliency map and the ground-truth saliency map.

Implementation Details: We train the network based on the training sey of DUTS dataset [23], which includes 10,553 images with pixel annotations. We augment the training set by horizontally flipping and rotating the images, in order to relieve the over-fitting problem. With a single GeForce GTX 1080Ti GPU, it takes about 13 hours to train the network, which converges after 8 epochs. The Adam optimizer [24] is used to train our model with an initial learning rate of 10^{-8}. The pre-trained VGG-16 net [20] is used as our initial network. We discard all the fully-connected layers and the last pooling layer in the VGG-16 net, so as to focus on the pixel task and to preserve the details in the last convolutional layer. We feed the input image, resized to 224×224 by bilinear interpolation, into the modified VGG-16 net to capture the initial deep multi-layer features. For other convolutional layers, we initialize the weights using the truncated normal method. The convolutional parameters of the MSFA module are not shared, and upsampling is performed simply by using bilinear interpolation. We set the number of iterations in Section 2.2 at 2. During testing, our proposed model runs about 27 fps with 224 × 224 resolution.

3.2. Comparison with State-of-the-art Methods

We compare our algorithm with ten state-of-the-art deep-learning-based algorithms, including MDF [25], ELD [6], Amulet [8], DSS [9], UCF [10]), NLDF [11], DCL [7], RFCN [14], GBR [12], MPFF [13]). For fair comparison, we use either the source code or the saliency maps provided by the repective authors.

Quantitative Evaluation. Figure 2 shows the PR curves of different approaches on the four datasets. We can observe from the PR curves that the proposed algorithm outperforms the other methods on the four datasets. Table 1 illustrates the performances of different methods, in terms of maximum F-measure and MAE. Our method can consistently outperform other approaches, across all the datasets with different measurements. It demonstrates the effectiveness of the proposed model. Meanwhile, our model also generally decreases MAE. This indicates that our model can estimate more accurate saliency maps than other state-of-the-art methods.

Qualitative Evaluation. Figure 3 shows the saliency maps generated by the proposed method, as well as the 10 state-of-the-art methods. It can be seen that our model can accurately detect salient objects, even in some challenging cases, e.g., complex boundary detection (the first row), multiple disconnected salient objects (the second row), objects near the image boundary (the third row) and low contrast between

Table 1. Quantitative comparison against 10 state-of-the-art methods using MAE and Max F-measure. The top three results are highlighted in red, green and blue, respectively.

Method	Year	ECSSD		HKU-IS		PASCAL-S		DUT-OMRON	
		F_β^{max}	MAE	F_β^{max}	MAE	F_β^{max}	MAE	F_β^{max}	MAE
MDF[25]	CVPR-2015	0.8316	0.1050	0.8605	0.1292	0.7636	0.1477	0.6944	0.0916
ELD[6]	CVPR-2016	0.8681	0.0726	0.8809	0.0726	0.7713	0.1233	0.7052	0.0910
Amulet[8]	ICCV-2017	0.8693	0.0588	0.8990	0.0521	0.8390	0.0993	0.7428	0.0976
DSS[26]	ICCV-2017	0.8985	0.0675	0.8876	0.0537	0.8185	0.1106	0.7462	0.0788
UCF[10]	ICCV-2017	0.9033	0.0691	0.8876	0.0612	0.8236	0.1158	0.7296	0.1204
NLDF[11]	CVPR-2017	0.9050	0.0626	0.9016	0.0480	0.8319	0.1007	0.7532	0.0796
DCL[7]	CVPR-2016	0.8871	0.0716	0.8818	0.0582	0.8087	0.1100	0.7175	0.0940
RFCN[14]	ECCV-2016	0.8981	0.0907	0.8947	0.0733	0.8375	0.1159	0.7387	0.0913
GBR[12]	ICME-2018	0.9093	0.0660	0.8926	0.0556	0.8289	0.1103	0.7584	0.0735
MPFF[13]	ICME-2018	0.9064	0.0646	0.8993	0.0513	0.8262	0.1072	0.7570	0.0756
Ours	-	0.9203	0.0550	0.9148	0.0435	0.8506	0.0907	0.7916	0.0677

Table 2. Assessment of individual modules on the ECSSD and DUT-OMRON datasets.

Model	ECSSD		DUT-OMRON	
	F_β^{max}	MAE	F_β^{max}	MAE
Baseline	0.9014	0.0646	0.7522	0.0795
Baseline+MSCA	0.9131	0.0586	0.7732	0.0667
Baseline+MLC	0.9082	0.0630	0.7711	0.0718
Baseline+MSCA+MLC	0.9203	0.0550	0.7916	0.0677

the objects and the backgrounds (the last one row).

3.3. Ablation Studies

The proposed framework is composed of two main modules, the MSCA module and the MLC module. To demonstrate the effectiveness of these modules, we also evaluate the performance of each of the modes. We only take out Conv blocks and Si in Figure 1 as the baseline model. And we embed MSCA or MLC into baseline respectively. Compared with the model without using MSCA, our model improves the maximum F-measure by 1.33% and 2.66% and decreases the MAE by 11.14% and 5.71% on ECSSD and DUT-OMRON datasets respectively. Compared with the model without MLFA, our model improves the maximum F-measure by 0.79% and 2.38% and decreases the MAE by 6.14% and -1.45% on ECSSD and DUT-OMRON datasets.

4. CONCLUSIONS

In this paper, we propose a novel multi-scale Capsule attention based Salient object detection deep Network. We first design the multi-scale capsule attention module to capture the features in multiple scales and make the network focus on the salient foreground cues. Then, multi-crossed skip-layer connections is utilized to aggregate features in different layers. The final saliency maps are generated by fusing the high-level semantic concept and low-level spatial details. Experimental results on four datasets demonstrate that our proposed approach outperforms 10 state-of-the-art methods under different evaluation metrics.

5. REFERENCES

[1] Zhixiang Ren, Shenghua Gao, Liang Tien Chia, and W. H. Tsang, "Region-based saliency detection and its application in object recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 5, pp. 769–779, 2014.

[2] Ali Borji, Simone Frintrop, Dicky N Sihite, and Laurent Itti, "Adaptive object tracking by learning background context," in *IEEE Computer Society Conference Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2012, pp. 23–30.

[3] Rui Zhao, Wanli Ouyang, and Xiaogang Wang, "Unsupervised salience learning for person re-identification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 3586–3593.

[4] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu, "Global contrast based salient region detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 569–582, 2015.

[5] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang, "Saliency detection via graph-based manifold ranking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 3166–3173.

[6] Gayoung Lee, Yu-Wing Tai, and Junmo Kim, "Deep saliency with encoded low level distance map and high level features," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 660–668.

[7] Guanbin Li and Yizhou Yu, "Deep contrast learning for salient object detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 478–487.

[8] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Xiang Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 202–211.

[9] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip Torr, "Deeply supervised salient object detection with short connections," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5300–5309.

[10] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Baocai Yin, "Learning uncertain convolutional features for accurate saliency detection," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 212–221.

[11] 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, vol. 2, p. 7.

[12] Xin Tan, Hengliang Zhu, Zhiwen Shao, Xiaonan Hou, Yangyang Hao, and Lizhuang Ma, "Saliency detection by deep network with boundary refinement and global context," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2018, pp. 1–6.

[13] Hengliang Zhu, Xin Tan, Zhiwen Shao, Yangyang Hao, and Lizhuang Ma, "Multi-path feature fusion network for saliency detection," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2018, pp. 1–6.

[14] Linzhao Wang, Lijun Wang, Huchuan Lu, Pingping Zhang, and Xiang Ruan, "Saliency detection with recurrent fully convolutional networks," in *European Conference on Computer Vision (ECCV)*, 2016, pp. 825–841.

[15] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton, "Dynamic routing between capsules," in *Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 3856–3866.

[16] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang, "Saliency detection via graph-based manifold ranking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 3166–3173.

[17] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia, "Hierarchical saliency detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 1155–1162.

[18] Rui Zhao, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang, "Saliency detection by multi-context deep learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1265–1274.

[19] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille, "The secrets of salient object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 280–287.

[20] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[21] Fisher Yu and Vladlen Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.

[22] Ali Borji, Ming-Ming Cheng, Huaizu Jiang, and Jia Li, "Salient object detection: A benchmark," *IEEE transactions on Image Processing*, vol. 24, no. 12, pp. 5706–5722, 2015.

[23] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 136–145.

[24] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[25] Guanbin Li and Yizhou Yu, "Visual saliency based on multiscale deep features," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 5455–5463.

[26] Tiantian Wang, Ali Borji, Lihe Zhang, Pingping Zhang, and Huchuan Lu, "A stagewise refinement model for detecting salient objects in images," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 4019–4028.