SALIENCY DETECTION VIA LOCAL SINGLE GAUSSIAN MODEL

Nan Xu, Yanqing Guo, Xiangwei Kong

School of Information and Communication Engineering, Dalian University of Technology, Dalian, China
E-mail: xunan@mail.dlut.edu.cn; {guoyq, kongxw}@dlut.edu.cn

ABSTRACT

Saliency detection has been long researched. However, most existing algorithms can not uniformly highlight salient objects. To approach this problem, we propose a novel saliency detection algorithm based on the Local Single Gaussian Model (LSGM). First, we utilize a bottom-up model to generate an initial saliency map and construct a background dictionary and a foreground dictionary based on the initial saliency map, respectively. Then, a LSGM is used to obtain a LSGM-based map. Note that we construct a corresponding LSGM for each superpixel region and thus the LSGM is a dynamic model with geometric structure information. Finally, we integrate the LSGM-based saliency map and the initial bottom-up map with global information as the final saliency map. Extensive experiments on four public datasets show that our algorithm outperforms state-of-the-art methods.

***Index Terms*—** Bottom-up model, local single Gaussian model, saliency map.

1. INTRODUCTION

Saliency detection is an important research domain of computer vision. It is beneficial for multitudinous applications, such as image and video compression, object detection and recognition, and content-aware image resizing, etc.

Generally speaking, there are mainly two categories of saliency detection algorithms: bottom-up models and top-down models. Top-down models [1] are task-driven and learn models from training samples with manually labeled ground truth. High-level cues and supervised learning frameworks are usually used to improve the algorithm performance. In contrast, bottom-up models are stimuli-driven and utilize some low-level features such as color and texture. In recent years, many bottom-up saliency detection models have been proposed. Itti *et al.*[2] propose a local center-surrounded contrasts method, which is highly influential and computational. Based on this method, many algorithms are developed accordingly, such as Ma and Zhang *et al.*[3], Achanta *et al.*[4]. Recently in [5], Cheng *et al.* propose another method based on soft image abstraction by taking both spatial distribution and appearance similarity of pixels into account. In [6], Liu *et al.* utilize a learning algorithm based on Partial Differential Equation to detect salient objects, and use a linear elliptic system with Dirichlet boundary to describe the relationships between seeds and other relevant points. In addition, sparse coding is also applied to saliency detection methods. In [7], Borji *et al.* first use the input image to train the dictionary, and then utilize sparse reconstruction errors to obtain the coefficient vector which denotes the feature of each image patch and is the basis to estimate saliency. Also, Li *et al.*[8] propose a saliency detection model via combining sparse and dense reconstruction errors. Recently, background-based model is utilized in saliency detection algorithms. Jiang *et al.*[9] propose a saliency detection method based on backgrounds, which utilizes the absorbed time in an absorbing Markov chain. Also, Li *et al.*[10] propose a saliency detection method based on a Gaussian background model. In this method, FT method [4] is used to generate an initial saliency map and then the author utilizes a Gaussian background model to calculate saliency values. Although the previous methods can detect salient objects accurately without considering image structure or local information, they are not able to highlight salient objects uniformly.

In this paper, in order to deal with the aforementioned problem, a saliency detection method based on a Local Single Gaussian Model is proposed. Different from most existing methods, we use a LSGM with local geometric structure information surrounding superpixel nodes to generate saliency maps. The contributions of our work are as follows:

- We apply a global bottom-up saliency map as an initial saliency map. Owing to the accuracy of the bottom-up saliency map, the final results are more reliable.
- We exploit the LSGM to construct saliency maps rather than using the global SGM which is an unchangeable model for each image. Contrarily, the LSGM is a dynamic model and each LSGM indicates the geometric structure information surrounding a superpixel node.
- We construct two LSGMs based on the background and foreground dictionaries, respectively, and then integrate them together to generate a LSGM-based saliency map.

This work is funded by the National Natural Science Foundation of China (Grant No. 61402079), the Foundation for Innovative Research Groups of the NSFC (Grant No. 71421001), the Open Project Program of the National Laboratory of Pattern Recognition (NLPR).

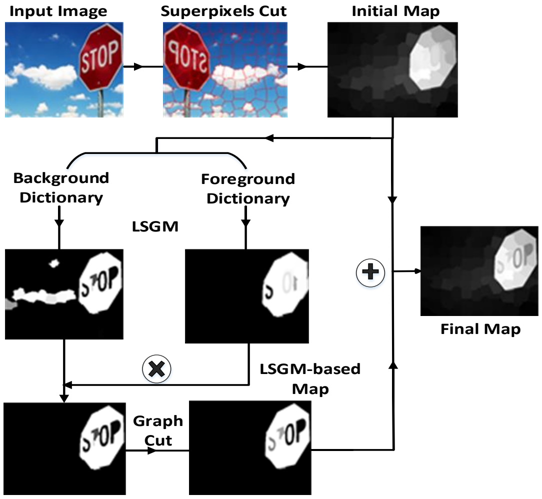

Fig. 1. Diagram of our proposed algorithm.

By utilizing the Area Under ROC Curve (AUC) and Precision and Recall (P-R) curve as two evaluation criterions, we evaluate our method and ten state-of-the-art methods on four public datasets. Experimental results show that our method performs favorably against the state-of-the-art methods.

2. THE PROPOSED APPROACH

To get the structural information, the input image is represented as a graph with superpixel nodes obtained by the SLIC method [11]. Each node denotes the average feature information of all pixels belonging to the corresponding superpixel node. Thus the significant structure information of the graph is retained. Next, we construct the background dictionary (BD) and the foreground dictionary (FD) based on the initial saliency map. Then, we utilize the LSGM to generate two kinds of saliency maps and integrate them together. Finally, we combine the initial saliency map and the LSGM-based map, which is an excellent integration of the global and local information. The algorithm flow chart is shown in Fig. 1.

2.1. Dictionary Construction

To obtain a more reliable dictionary, we utilize the bottom-up saliency map provided in [12] as a rough initial map which is denoted as S_b. Note that the LLC method in [12] describes the geometry of feature space as a combination of superpixel nodes while ignoring the geometric structure surrounding them. To address this problem, for each node, we construct the LSGM by calculating the mean and covariance matrix of other nodes neighboring it, which takes the geometric structure surrounding each node into account and can highlight salient objects uniformly.

A SGM-based model requires a reliable dictionary for computing saliency values. In this paper, in order to avoid the boring manual labeling process, we use the bottom-up saliency map S_b to construct two dictionaries, namely background dictionary and foreground dictionary. First, we can get the binary value of superpixel nodes SN by scanning all nodes of the input image, which could be denoted as:

$$SN\left(i\right) = \begin{cases} 0 & S_b\left(i\right) < \lambda_1 \\ 1 & S_b\left(i\right) > \lambda_2 \end{cases} \qquad i = 1, 2, ..., t, \tag{1}$$

where i indexes the superpxiel node of the input image, t is the number of all superpixel nodes. Also, the dictionaries are not sensitive to λ_1, so λ_1 is a fixed value for all images and is set as 0.05. λ_2 is an adaptive value and is set as T times of S_{bavg}, where S_{bavg} is the average value of S_b, and T is set as 1.25 empirically. Then, we define the superpixel nodes corresponding to zero elements and non-zero elements of SN as background nodes and foreground nodes, respectively. Finally, using the LAB and RGB color spaces which have the complement effects [7] on the experimental results, we can extract the 6 dimensional feature vectors of background and foreground nodes to construct BD and FD, respectively.

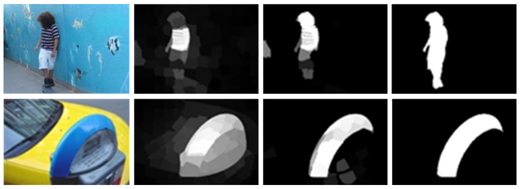

Fig. 2. Saliency maps. From left to right: Input images, global SGM maps, LSGM maps, ground truth.

2.2. Local Single Gaussian Model Construction

To highlight salient objects uniformly, we exploit the LSGM to construct saliency maps. For each superpixel region, we construct a corresponding LSGM which is a dynamic model with local geometric structure information.

1) Single Gaussian Model: Generally speaking, Gaussian model can be formulated as:

$$G(x) = \sum_{l=1}^{k} \frac{\alpha_l}{\sqrt{2\pi\left|\Sigma_l\right|}} \exp[-\frac{1}{2}(x-\mu_l)^T \Sigma_l^{-1}(x-\mu_l)], \tag{2}$$

where u_l, Σ_l are the mean and covariance matrix, α_l is the weight of the l-th component and k is the total number of components. In this paper, k is set as 1, namely Single Gaussian Model (SGM). Now let $D = [d_1, d_2, ..., d_i, ..., d_n] \in \mathbb{R}^{m \times n}$ be the dictionary samples, where d_i indicates the feature vector of m (m=6) dimensions and n is number of samples. The dictionary D is used to train a SGM. Then we calculate the probability distribution of each node belonging to the SGM:

$$F(r_i) = \frac{1}{\sqrt{2\pi|\Sigma|}} \exp[-\frac{1}{2}(f_i-\mu)^T \Sigma^{-1}(f_i-\mu)], \tag{3}$$

where f_i is the 6 dimensional feature for each node r_i, μ and Σ denote the mean and covariance matrix of the dictionary D, respectively.

2) Local Single Gaussian Model: In this paper, we can construct a SGM named Local Single Gaussian Model (LSGM) for each node r_i. We select the K nearest neighbors in

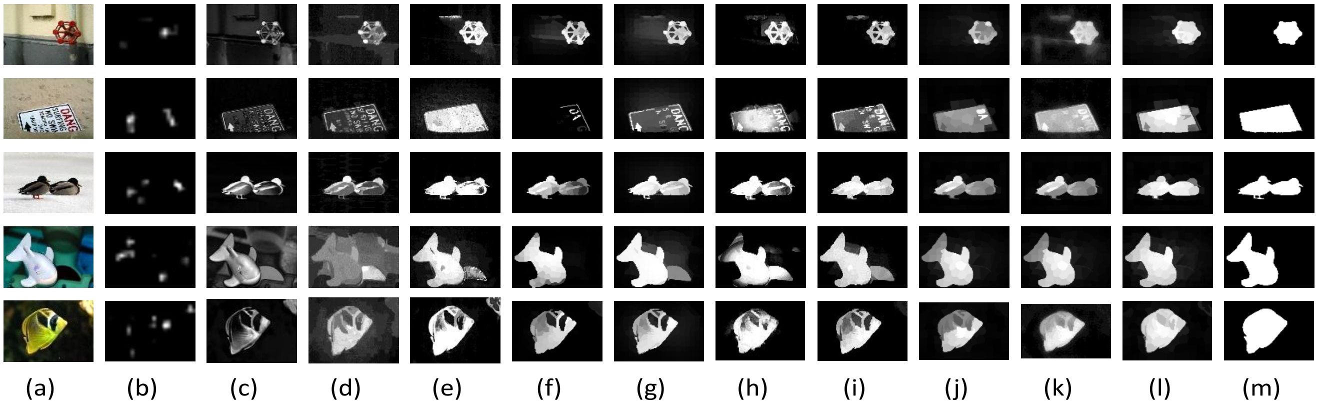

Fig. 3. Saliency maps. From left to right: (a) input images, (b) IT [2], (c) FT [4], (d) RC [13], (e) XL11[14], (f) GMR13 [15], (g) AMC [9], (h) DSR13[8], (i) wCO14 [16], (j) Tong15 [12], (k) BL15[17], (l) Our method and (m) ground truth.

the spatial domain from the original dictionary as the local dictionary D_i for each node r_i, where $K = 2n/5$ empirically and n is the number of the original dictionary samples. The dictionary D_i is used to train a LSGM. Then we calculate the probability distribution of each node belonging to the LSGM:

$$P(r_i) = \frac{1}{\sqrt{2\pi|\Sigma_i|}} \exp[-\frac{1}{2}(f_i - \mu_i)^T \Sigma_i^{-1}(f_i - \mu_i)], \quad (4)$$

where f_i is the 6 dimensional feature of the superpixel node r_i, μ_i and Σ_i are the mean and covariance matrix of the local dictionary D_i, respectively. Thus for each superpixel node, we can get a corresponding LSGM. As shown in Fig. 2, compared with the global SGM, the LSGM is effective and can generate more reliable backgrounds.

3) Reinforcement: In Section 2.1, we obtain two kinds of dictionaries, BD and FD.

When BD is used, the saliency map can be formulated as:

$$S_t^b(r_i) = \frac{1}{2\sigma^2} \exp(-\frac{P^b(r_i)}{2\sigma^2}), \quad (5)$$

When FD is used, the saliency map can be formulated as:

$$S_t^f(r_i) = \frac{1}{2\sigma^2} (1 - \exp(-\frac{P^f(r_i)}{2\sigma^2})), \quad (6)$$

where σ is set as 12 empirically. $P^b(r_i)$ and $P^f(r_i)$ denote the background and foreground probability models, respectively. $S_t^b(r_i)$ and $S_t^f(r_i)$ denote the normalized saliency value of the node r_i utilizing BD and FD, respectively.

BD-based LSGM can generate a more reliable background. In contrast, FD-based LSGM can detect a foreground. Thus we can obtain a more accurate saliency map by integrating two saliency maps based on two kinds of dictionaries, and the formula is as follows:

$$S_t(r_i) = S_t^b(r_i) \times S_t^f(r_i), \quad (7)$$

where $S_t(r_i)$ denotes the saliency value of the superpixel node r_i based on the LSGM. Additionally, in this paper, we use the Graph Cut method [18] to smooth the pixel-wise maps provided by the equation (7) and denote the LSGM-based saliency map as S_t.

2.3. Final Saliency Map

The LSGM-based saliency map S_t denotes effective local structure information and previous bottom-up saliency map S_b denotes effective global information. Thus we combine S_b and S_t together, and the formula is as follows:

$$S = w \times S_{\mathrm{b}} + (1 - w) \times \mathrm{S}_t, \quad (8)$$

where w is a balance factor and $w = 0.7$ to weigh the bottom-up map more than the LSGM-based map according to experimental results.

3. EXPERIMENTS

We utilize the P-R curve and AUC value as the evaluation criterions to compare the proposed algorithm with ten state-of-the-art methods on four standard datasets. The ECSSD dataset [19] consists of 1000 images with complex scenes from the Internet. Thus, it is more challenging. The SOD dataset [20] contains 300 images from the Berkeley segmentation dataset. Many images have multiple salient objects with various locations and sizes. The MSRA5000 dataset [21] contains 5000 images which are labeled with pixel-wise ground truth. Also, we use the THUS dataset [22] which consists of 10,000 images labeled with pixel-wise ground truth.

The ten methods are IT [2], FT [4], RC [13], XL11[14], GMR13 [15], AMC [9], DSR13 [8], wCO14 [16], Tong15 [12], BL15 [17]. The wCO14 method only provides saliency maps on the THUS dataset and the RC method doesn't provide saliency maps on the ECSSD dataset. We perform all experiments by using the MATLAB tool on a computer with Intel Xeon E5-2650 v2CPU (2.6GHz) and 32G RAM.

3.1. Experimental Results Compared With 10 Methods

In this section, the comparative results of 10 methods are showed in terms of qualitative and quantitative evaluations.

1) Qualitative Evaluation: Fig. 3 demonstrates the samples of saliency maps obtained by our method and the other ten methods. The results show that our saliency maps and the

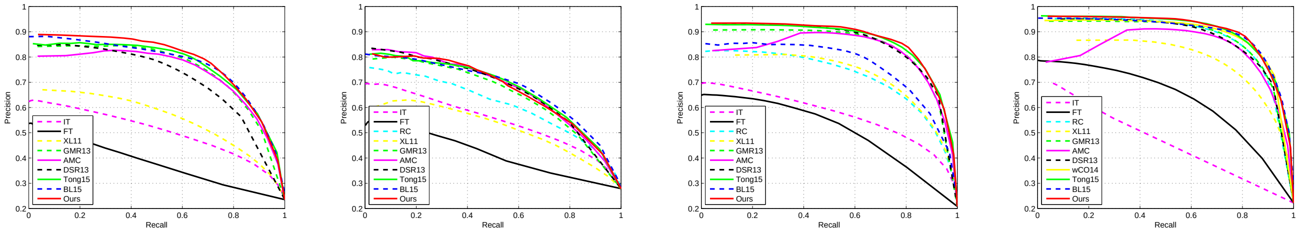

Fig. 4. Experimental results. From left to right: Experimental results on the ECSSD, SOD, MSRA5000 and THUS dataset.

Table 1. AUC values on the ECSSD, SOD, MSRA5000 and THUS datasets. The best two results are shown in red and blue.

	IT	FT	RC	XL11	GMR13	AMC	DSR13	wCO14	Tong15	BL15	Ours
ECSSD	.7920	.6296	-	.8135	.8827	.9079	.8619	-	.9117	.9142	.9162
SOD	.7838	.5974	.7924	.7597	.7899	.8391	.8210	-	.8366	.8466	.8473
*MSRA*5000	.8504	.7363	.8951	.9098	.9261	.9476	.9382	-	.9544	.9534	.9588
THUS	.6169	.7849	.9357	.9243	.9283	.9464	.9369	.9437	.9615	.9622	.9638

ground truth are very similar, which demonstrates our method is very favorable over the experimental methods. In addition, we obtain a LSGM-based saliency map with each LSGM corresponding to a superpixel node, which takes the local structure information into account. Thus compared with the other state-of-the-art methods, our method is able to locate the saliency objects precisely, highlight the saliency objects uniformly, and suppress background noises effectively.

2) Quantitative Evaluation: To evaluate the performance quantitatively, we utilize the P-R curve and the AUC value to evaluate our algorithm and the other state-of-the-art methods. Firstly we use the P-R curve as the evaluation criterion to evaluate these methods on four public datasets. As shown in Fig. 4, the experimental results demonstrate that the superior performance of our method on the larger datasets (ECSSD, MSRA5000 and THUS). On the SOD dataset, the P-R curve of our method is very close to the other methods. Then we utilize the AUC value as another evaluation criterion which is reliable as much as the P-R curve. As shown in Table 1, the AUC values of our method are the highest on four public datasets, which demonstrates our method outperforms previous state-of-the-art methods in terms of AUC values. In summary, experimental results demonstrate that superior performance of our method over previous state-of-the-art methods.

3.2. Validation of Each Section in the Proposed Method

The proposed algorithm consists of two components which are bottom-up saliency map and LSGM-based saliency map. Fig. 5 shows the P-R curves of two components and our method on the ECSSD dataset. The experimental results demonstrate that the LSGM method is effective and contributes to the final results.

- The P-R curve of the final result is higher than that of bottom-up model and LSGM, respectively, which demonstrates that the LSGM method has the superior performance by optimizing the initial bottom-up saliency maps.
- The contributions of the bottom-up model and the LSGM are different. The P-R curve of the bottom-up model plays a more important role than that of the LSGM, so bottom-up saliency maps have a larger weight. The weights are obtained by multiple experiments.

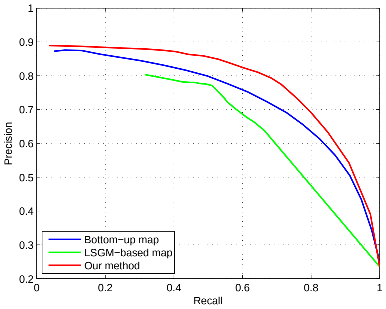

Fig. 5. Results on the ECSSD dataset for two components of our method.

4. CONCLUSION

In this paper, we design a dynamic model for saliency detection by utilizing the LSGM and integrate the global and local information by fusing the initial saliency map and the LSGM-based saliency map. In this work, we construct a corresponding LSGM for each superpixel node. Additionally, we combine two LSGMs based on the background and foreground dictionaries to optimize the LSGM-based saliency map. We compare our method with other ten methods on four datasets and experimental results show that our method is able to detect saliency accurately, suppress the background noises effectively, and uniformly highlight the salient objects. Moreover, the quantitative results also demonstrate that our method outperforms state-of-the-art methods.

5. REFERENCES

[1] J. Yang and M.-H. Yang, "Top-down visual saliency via joint crf and dictionary learning," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition (CVPR)*, Providence, RI, Jun. 2012, pp. 2296–2303.

[2] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[3] Y. Ma and H. Zhang, "Contrast-based image attention analysis by using fuzzy growing," in *Proc. ACM Int. Conf. Multimedia*, Berkeley, California, Nov. 2003, pp. 374–381.

[4] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition (CVPR)*, Miami, FL, Jun. 2009, pp. 1597–1604.

[5] M.-M. Cheng, J. Warrell, W.-Y. Lin, S. Zheng, V. Vineet, and N. Crook, "Efficient salient region detection with soft image abstraction," in *Proc. IEEE Int. Conf. Comput. Vision (ICCV)*, Sydney, Australia, Dec. 2013, pp. 1529–1536.

[6] R. Liu, J. Cao, Z.-C Lin, and S.-G Shan, "Adaptive partial differential equation learning for visual saliency detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition (CVPR)*, Columbus, OH, Jun. 2014, p-p. 3866–3873.

[7] A. Borji and L. Itti, "Exploiting local and global patch rarities for saliency detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition (CVPR)*, Providence, RI, Jun. 2012, pp. 478–485.

[8] X. Li, H.-C. Lu, L. Zhang, X. Ruan, and M.-H. Yang, "Saliency detection via dense and sparse reconstruction," in *Proc. IEEE Int. Conf. Comput. Vision (ICCV)*, Sydney, Australia, Dec. 2013, pp. 2976–2983.

[9] B.-W. Jiang, L.-H. Zhang, H.-C. Lu, and M.-H. Yang, "Saliency detection via absorbing markov chain," in *Proc. IEEE Int. Conf. Comput. Vision (ICCV)*, Sydney, Australia, Dec. 2013, pp. 1665–1672.

[10] J.-L. Li, F. Meng, and Y.-C. Zhang, "Saliency detection using a background probability model," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Quebec City, Canada, Sept. 2015, pp. 2189–2193.

[11] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Ssstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[12] N. Tong, H.-C. Lu, Y. Zhang, and X. Ruan, "Salient object detection via global and local cues," *Pattern Recognition*, vol. 48, no. 10, pp. 3258–3267, Oct. 2015.

[13] M.-M. Cheng, G.-X. Zhang, N.J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition (CVPR)*, Providence, RI, Jun. 2011, pp. 409–416.

[14] Y. Xie and H. Lu, "Visual saliency detection based on bayesian model," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Brussels, Belgium, Sept. 2011, pp. 653–656.

[15] C. Yang, L.-H. Zhang, H.-C. L, and M.-H. Y, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition (CVPR)*, Portland, Oregon, Jun. 2013, pp. 3166–3173.

[16] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition (CVPR)*, Columbus, OH, Jun. 2014, pp. 2814–2821.

[17] N. Tong, H.-C. Lu, X. Ruan, and M.-H. Yang, "Salient object detection via bootstrap learning," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition (CVPR)*, Boston, MA, Jun. 2015, pp. 1884–1892.

[18] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.

[19] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition (CVPR)*, Portland, Oregon, Jun. 2013, pp. 1155–1162.

[20] V. Movahedi and J.-H. Elder, "Design and perceptual validation of performance measures for salient object segmentation," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, San Francisco, CA, Jun. 2010, pp. 49–56.

[21] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 353–367, Feb. 2011.

[22] M.-M. Cheng, N.J. Mitra, X. Huang, P.H. Torr, and S.-M. Hu, "Salient object detection and segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 2, no. 3, 2011.