Research Article

Saliency Detection via Fusing Color Contrast and Hash Fingerprint

Yin Lv, Xuanrui Zhang, and Yong Wang ⓘ

School of Mechanical Engineering and Electronic Information, China University of Geosciences, Wuhan 430074, China

Correspondence should be addressed to Yong Wang; wy112708@163.com

Saliency detection is a technique for automatically extracting regions of interest from the background and has been widely used in the computer vision field. This study proposes a simple and effective saliency detection method combining color contrast and hash fingerprint. In our solution, the input image is segmented into nonoverlapping superpixels, so as to perform the saliency detection at the region level to reduce computational complexity. A background optimization selection is used to construct an accurate background template. Based on this, a saliency map that highlights the whole salient region is obtained by estimating color contrast. Besides, another saliency map that enhances the salient region while restraining the background is also generated through hash fingerprint matching. Ultimately, the final saliency map can be obtained by fusing the two saliency maps. Comparing the performance with other methods, the proposed algorithm works better even in the presence of complex background or very large salient regions.

1. Introduction

Saliency detection can quickly screen important information so as to locate objects of interest in an image. It has been widely used in many applications, such as target detection [1], target recognition [2], image compression [3], and image retrieval [4]. Existing saliency detection methods can be roughly divided into top-down models and bottom-up models [5, 6]. The top-down models generally utilize high-level features or supervised learning methods to measure the saliency. For example, Liu et al. [7] proposed a binary saliency estimation model by training a conditional random field to combine a set of novel features. Wang et al. [8] used a trained classifier called the auto-context model for salient region detection. Lu et al. [9] generated a set of salient seeds through learning and used the seeds to identify the salient regions. Dai et al. [10] extracted high-level semantic features through a region-based fully convolutional network so as to obtain the saliency map. Jiang et al. [11] used the supervised learning approach to map the regional feature vector to a saliency score and fused the saliency scores across multiple levels to yield the saliency map. Although these kinds of models achieve good detection performance, the calculations are comparatively expensive and time consuming.

Comparatively, the bottom-up models primarily exploit low-level cues such as texture, intensity, color, and edge to measure saliency. Thus, this kind of method provides simpler and faster solutions and is more widely used in saliency detection. Itti et al. [12] presented a saliency map by combining three feature maps including color, intensity, and orientation at different scales. Achanta et al. [13] proposed a frequency-tuned approach to estimate the contrast difference between the mean image feature vector and the corresponding image pixel vector value in the Gaussian blurred version. Zhai and Shah [14] calculated the saliency using the sum of the color distance between the pixel and all the other pixels in the image. Achanta and Süsstrunk [15] estimated saliency using the contrast between the color and luminance features of each pixel and its maximum symmetry surrounding area. These methods have simple calculations but suffer great difficulty in handling images with complex backgrounds. To solve this problem, some algorithms introduced background prior and treated the image boundary region as the background for saliency detection. Wei et al.

[16] proposed a saliency measure called geodesic saliency that uses boundary and connectivity priors to provide more clues for the saliency calculation. Yang et al. [17] determined the saliency by the similarity of the image elements (pixel or region) with foreground cues or background. However, these approaches naively select the image border as the background so that they may fail when salient regions locate along the image boundary. Therefore, some research made further optimization in order to improve the detection performance. For instance, Zhu et al. [18] proposed a robust background measure based on boundary connectivity to constrain background regions and obtained salient regions using foreground cues. Besides, in our earlier work [19], an accurate background template was first constructed, and based on it the saliency map was obtained through image sparse representation and color features combination.

Aiming at the abovementioned problems, we propose a novel and effective salient detection algorithm to highlight salient regions while restraining the image background. First, simple linear iterative clustering (SLIC) [20] is utilized to segment the original image into superpixels, and the subsequent operations are carried out based on each superpixel, which helps to reduce computation complexity. Then, the accurate background template is generated through background optimization, which can improve the accuracy of subsequent saliency detection. The first saliency map corresponding to the original image is obtained by calculating the color contrast between superpixels based on the background template. On the other hand, the original image is divided into blocks of equal size, and the hash fingerprint of each block is calculated based on discrete cosine transformation (DCT). Thus, the second saliency map is obtained by matching hash fingerprints of image blocks. Finally, the two saliency maps are fused to generate the final saliency map. The framework of the proposed algorithm is shown in Figure 1.

The main contributions of this study are summarized as follows. First, we design a background optimization scheme to ensure that salient regions can be completely detected even when they are situated at the boundary of the image. Second, we use hash fingerprint matching to rapidly extract salient regions and well suppress the background. Third, we propose a fusion strategy to produce the final saliency map. Comprehensive experiments demonstrate that the proposed method has desirable detection accuracy.

2. Saliency Detection

2.1. Background Optimization Selection. To retain more structural information about the original image, we first utilize the SLIC algorithm to segment the image into nonoverlapping superpixels. Besides, the number of superpixels is much smaller than that of the original pixels, which helps increase the calculation speed. Most of the existing methods based on SLIC usually choose the edge superpixels as the background for saliency detection. However, in some cases, the salient regions usually extend to the edge of the image. If the edge superpixels of the image are

directly labeled as the background, it might cause the subsequent saliency detection to be degraded.

To address this issue, an optimization scheme is designed to generate a refined background template. After the image is divided into superpixels, all outermost superpixels are chosen to build the candidate background template $\mathbf{B}$, consisting of n superpixels. If a certain superpixel in $\mathbf{B}$ belongs to the salient region, there will be a big difference in color features between it and other superpixels. Moreover, the larger the color difference is, the more likely it belongs to the salient region. While considering the effect of the spatial distance between superpixels, for each superpixel in $\mathbf{B}$, we calculate the spatially weighted sum of color distances between it and all other superpixels and then judge whether it belongs to the background. Specifically, for superpixel i, the spatially weighted sum is denoted by

$$S_B(i) = \sum_{j=1}^{n} d_c(i, j) \cdot \exp -\left(\frac{d_s^2(i, j)}{2\sigma^2}\right), \quad (1)$$

where $d_c(i, j)$ and $d_s(i, j)$ are, respectively, the color distance and the spatial distance between two superpixels i and j, σ is empirically set to 0.25. Then the superpixels whose spatially weighted sum is greater than a specific threshold α will be eliminated from $\mathbf{B}$ to obtain an accurate background template. The threshold should be determined adaptively according to the input image, and it is defined as $\alpha = S_{B\max} - \lambda \cdot D$, where λ is a constant factor and set to 8 for the best performance, $S_{B\max}$ and D are the largest value, and the variance of $S_B(i)_{i=1...n}$, respectively. This means that only certain outermost superpixels are selected to construct the final background template $\mathbf{B}_{\mathrm{opt}}$. More details about background optimization selection are available in our earlier work [19].

2.2. Color Contrast-Based Saliency Map. Naturally, the more a superpixel varies in color from its surrounding regions, the larger the saliency value of the superpixel. Also, the closer the spatial distance between the superpixel and its surrounding superpixels, the more the contribution of the color contrast between them to the saliency value. Thus, we use color contrast weighted by the distances between a superpixel and other superpixels to measure the saliency value of the superpixel. Assume that the rest region of the image is denoted as $\mathbf{F}$ and contains m superpixels after removing superpixels belonging to $\mathbf{B}_{\mathrm{opt}}$. For the superpixel i in $\mathbf{F}$, the saliency value $S_F(i)$ can be formulated in a similar way to (1), the only difference is that superpixels used for calculation belong to $\mathbf{F}$.

To further highlight the salient region, the saliency of each superpixel in $\mathbf{F}$ is enhanced by integrating the color contrast between the superpixel and the background template. Specifically, the modified saliency value of the superpixel i is denoted by

$$S_f(i) = S_F(i) \cdot \left\| \mathbf{c}_i - \mathbf{c}_a \right\|_{nrm}, \quad (2)$$

where $\mathbf{c}_i$ is the color vector of superpixel i in CIELab space, $\mathbf{c}_a$ is the average color vector of all superpixels in $\mathbf{B}_{\mathrm{opt}}$,
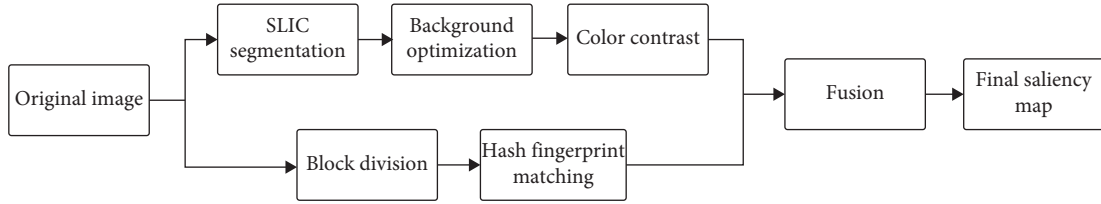
FIGURE 1: The framework of the proposed algorithm.

and $||\cdot||_{nrm}$ denotes the normalized Euclidian distance, that is the color contrast is normalized to [0, 1].

Figure 2 shows example results of the background optimization and the color contrast-based saliency map. It can be seen in the Figures 2(c) and 2(d), the background optimization selection can refine the background template. The outermost superpixels corresponding to the salient object are eliminated from the candidate background template. Besides, it is clear that the color contrast-based saliency map illustrated in Figure 2(e) can highlight the whole salient region well, but at the same time may not effectively suppress some background regions. Therefore, it is essential to perform further operations to improve the detection results.

2.3. Hash Fingerprint-Based Saliency Map.

Background usually contains more low-frequency information compared to the salient region. The hash algorithm based on DCT can effectively make use of the low-frequency information in image blocks, and the hash fingerprints can measure the similarity between image blocks. Therefore, hash fingerprint matching can be used to estimate the saliency of each image block and meanwhile better suppress the background.

In order to extract the hash fingerprint, the input image is first converted to a grayscale image and then divided into b nonoverlapping blocks of size 8×8. Note that the image should be resized by interpolation if its width and height are not integer multiples of 8. For each block, we can obtain the 8×8 coefficient matrix by applying a DCT. Next, we traverse the coefficient matrix using the mean of all coefficients as the threshold, meanwhile setting the coefficients less than the mean to zero and other coefficients to 1. Finally, we convert the 8×8 matrices made up of zeros and ones into a 64×1 flattened vector using a zigzag scan so as to obtain the hash fingerprint of the image block. Based on this, the saliency is represented as the spatially weighted sum of hamming distance. Thus, for image block i, the saliency value is calculated by

$$S_h(i) = \sum_{j=1}^{b} d\left(h_i, h_j\right) \cdot \exp\left(-\frac{(x-u)^2 + (y-v)^2}{8}\right), \quad (3)$$

where (x, y) and (u, v) are separately the coordinates of block i and another block j, h_i and h_j correspond to the hash fingerprints of the two blocks, and $d(h_i, h_j)$ represents the hamming distance between h_i and h_j.

Examples of hash fingerprint-based saliency detection are illustrated in Figure 3, where Figures 3(a) and 3(b) show the original image and the image blocks of size 8×8, and Figure 3(c) is the saliency map. Compared to Figure 2(e), it

can be seen from Figure 3(c) that the detection results can effectively remove most of the background. Nevertheless, there are two problems with this solution. The first is the outline of the saliency map is not too clear due to blocking artifacts, and this is because that the image is divided into small blocks and each block is characterized as a saliency value. The second is the extracted salient object is not complete and uniform enough. The reason behind this is that the edge region of the salient object that has a relatively large saliency value is highlighted, whereas other regions are just the opposite.

2.4. Saliency Map Fusion.

As described above, color contrast-based saliency detection has predominance in highlighting the whole salient region, while hash fingerprint-based saliency detection has an advantage in background suppression. Therefore, it is feasible to combine the two to generate a final saliency map. Inspired by the idea proposed in [21], the final saliency map can be obtained by retaining and balancing the respective advantages of the two saliency maps. Specifically, the final saliency map is generated by using the hash fingerprint-based saliency map to refine the color contrast-based saliency map, and the formula is described as follows:

$$S = S_f \cdot \left(1 - \exp\left(-\gamma \cdot S_h\right)\right), \quad (4)$$

where the constant factor γ is set to 5, which is used to control the influence degree of S_h on the final saliency map. It is clear that, when S_f remains unchanged, the final saliency value decreases with the decrease of S_h. As a result, the saliency value of the background region will be small because the corresponding S_h is usually very small. This means that the background can be suppressed effectively via the saliency map fusion, as shown in Figure 4(d).

3. Experiment Results

In this section, we evaluate the performance of the proposed method and compare it with eight state-of-the-art saliency detection algorithms, including FT [13], IT [12], GBMR [17], CA [22], LC [14], MSS [15], SF [23], and our earlier work [19] named SRFC. These methods are selected for comparison because they belong to bottom-up models and are the most classical saliency detection algorithms. Moreover, the executable codes for the above methods are publicly available.

In the experiments, five representative public datasets, DUTS [24], ECSSD [25], MSRA10K [26], SOD [27], and
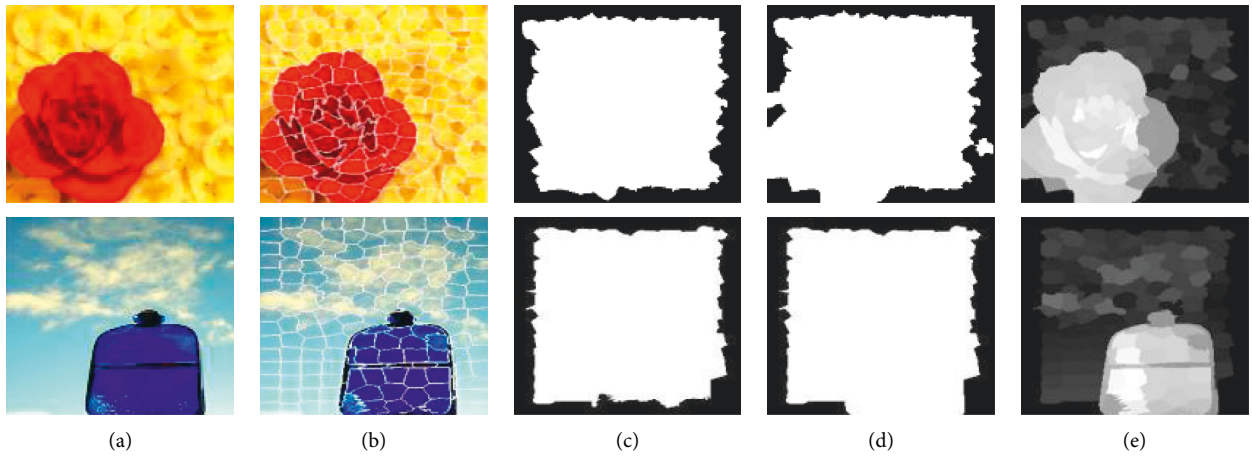
(a) (b) (c) (d) (e)

FIGURE 2: Color contrast-based saliency detection. (a) Original image; (b) superpixel segmentation; (c) candidate background template; (d) final background template; and (e) color contrast-based saliency map.
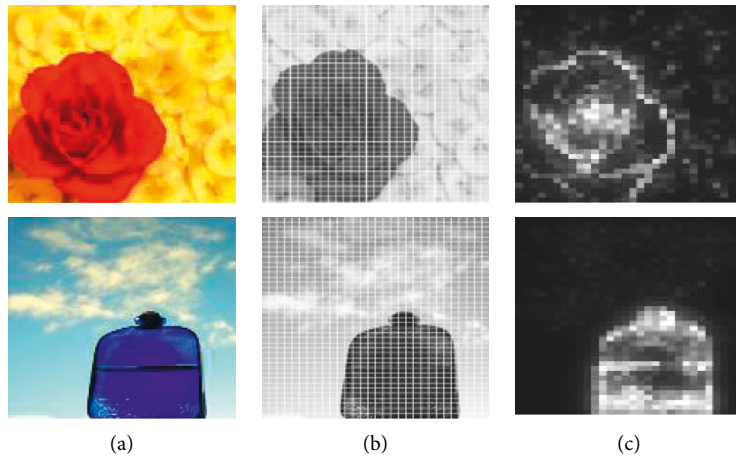

(a) (b) (c)

FIGURE 3: Hash fingerprint-based saliency detection. (a) Original image; (b) image division; and (c) hash fingerprint-based saliency map.
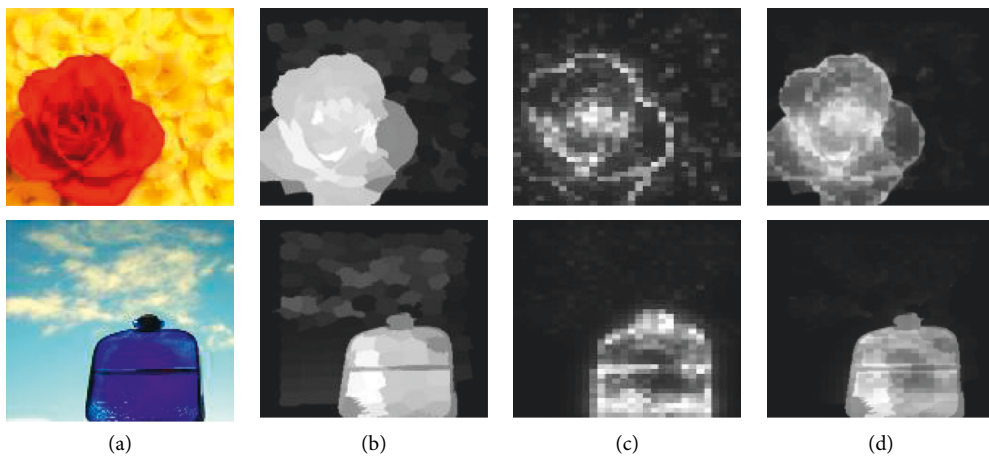

(a) (b) (c) (d)

FIGURE 4: Saliency fusion. (a) Original image; (b) color contrast-based saliency map; (c) hash fingerprint-based saliency map; and (d) final saliency map.
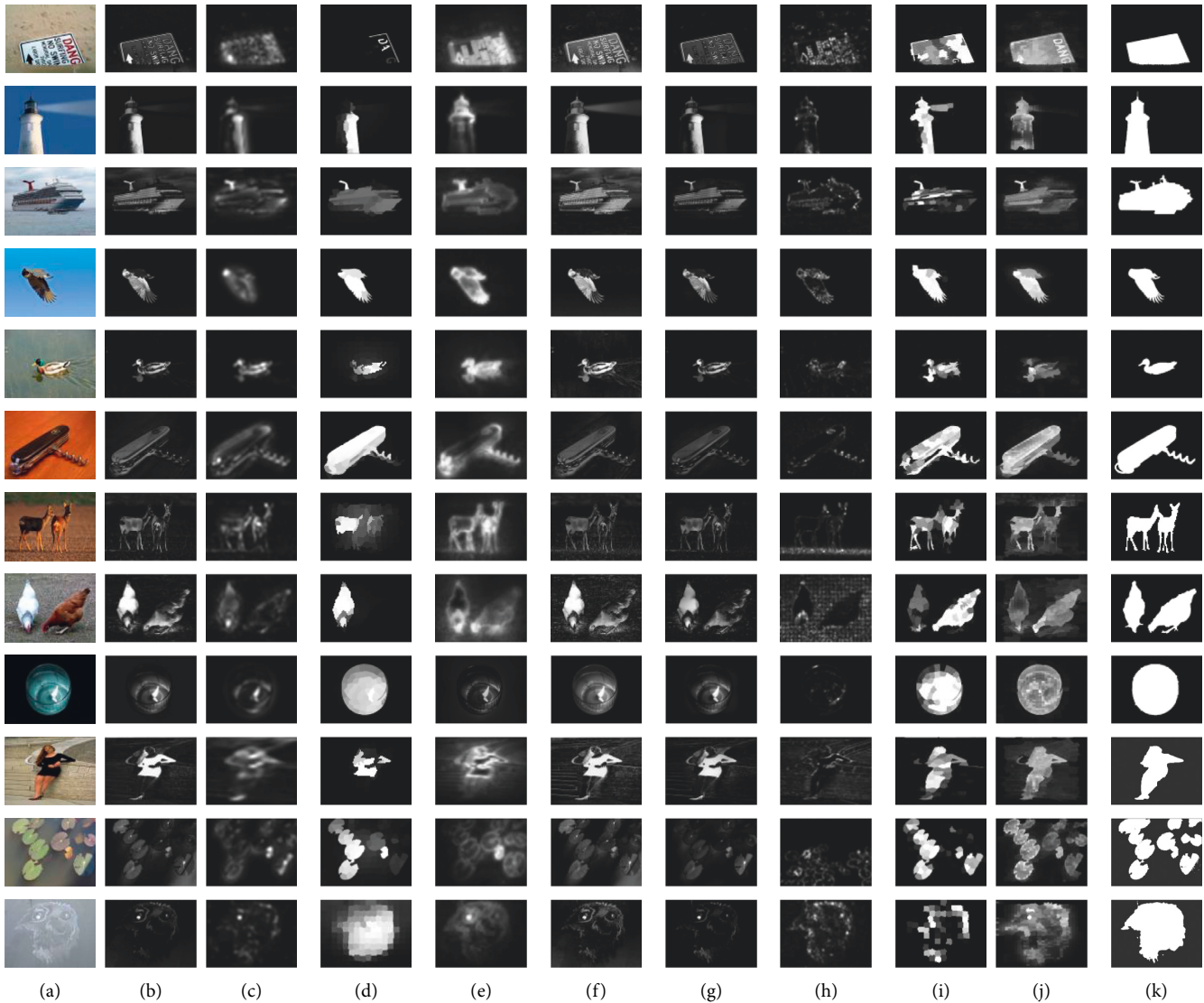
FIGURE 5: Saliency detection using different algorithms over six datasets. (a) Original image; (b) FT; (c) IT; (d) GBMR; (e) CA; (f) LC; (g) MSS; (h) SF; (i) SRFC; (j) ours; and (k) ground truth.

HKU_IS [28], and a self-made dataset DADA-W are adopted to compare various saliency detection methods. Note that DADA-W is an image dataset that we captured from various water environments in which the salient regions include garbage, algae, spilled oil, etc. We randomly choose 60 images from each of the six datasets for performance testing. Due to space constraints, only a few representative images are selected and the comparison results are shown in Figure 5, where Figure 5(a) is the original image, Figures 5(b)~5(j) are the saliency maps using different algorithms, and Figure 5(k) is the ground truth.

It can be seen from Figure 5 that the saliency maps of the proposed method are more complete and have fewer error detections in most cases. Specifically, our method can not only suppress the background but also highlight the whole salient region. Especially for complex scenarios, the proposed method and SRFC obviously outperform the others because they both adopt the background optimization selection. For example, for the 11th image, all the state-of-the-art methods except for the two methods we proposed can hardly completely extract lotus leaves from the water. Besides, compare to SRFC, the proposed method is more effective and can completely extract the whole salient object from the background. For certain images, there maybe exist small holes and fragmentation in the detection results of SRFC, as seen from the results of the 3rd and 9th images. The reason is that the goal of SRFC is to distinguish the salient region from the background as far as possible, whereas the proposed method puts much emphasis on highlighting the whole salient region while suppressing the background.

Three standard metrics, namely the area under ROC curve (AUC) score, mean absolute error (MAE) score, and $F_{\beta\text{-score}}$ ($\beta = 0.3$) are utilized to quantitatively compare the

TABLE 1: Quantitative comparison of different methods over six image datasets.

Dataset	Metrics	Method								
		FT	IT	GBMR	CA	LC	MSS	SF	SRFC	Ours
MASK10k	AUC	0.8546	0.9144	0.9261	0.9504	0.8762	0.9326	0.9023	0.8952	0.9780
	MAE	0.1972	0.1929	0.1303	0.1787	0.1829	0.1851	0.2055	0.1045	0.1265
	$F_{\beta\text{-score}}$	0.8438	0.7818	0.9335	0.7789	0.8459	0.8840	0.7402	0.9224	0.9494
SOD	AUC	0.7175	0.8888	0.9454	0.9023	0.7212	0.8725	0.8745	0.8372	0.9472
	MAE	0.3284	0.3120	0.2794	0.2927	0.3214	0.3171	0.3423	0.2476	0.2427
	$F_{\beta\text{-score}}$	0.8136	0.8137	0.9636	0.8136	0.7993	0.8720	0.7991	0.8691	0.9254
DUTS	AUC	0.8274	0.8836	0.9200	0.8973	0.8277	0.8840	0.7974	0.8866	0.9679
	MAE	0.2465	0.2578	0.1817	0.2458	0.2439	0.2429	0.2708	0.1447	0.1752
	$F_{\beta\text{-score}}$	0.8358	0.7582	0.9298	0.7810	0.8257	0.8480	0.6805	0.9313	0.9354
ECSSD	AUC	0.7250	0.8702	0.9075	0.8875	0.7124	0.8557	0.8449	0.8266	0.9248
	MAE	0.3006	0.2635	0.2311	0.2458	0.2938	0.2623	0.2820	0.2025	0.2184
	$F_{\beta\text{-score}}$	0.7651	0.7556	0.9182	0.7989	0.7569	0.8315	0.7476	0.8732	0.8756
HKU_IS	AUC	0.8110	0.8887	0.8483	0.8839	0.8007	0.8886	0.6379	0.9244	0.9342
	MAE	0.2019	0.2074	0.1868	0.2422	0.2028	0.1823	0.2387	0.0881	0.1590
	$F_{\beta\text{-score}}$	0.7554	0.6986	0.8388	0.6704	0.7365	0.7922	0.4673	0.9139	0.8600
DADA-W	AUC	0.8406	0.9477	0.9374	0.9443	0.8041	0.9065	0.9192	0.8664	0.9610
	MAE	0.1538	0.1350	0.1185	0.1569	0.1611	0.1410	0.1418	0.1504	0.1147
	$F_{\beta\text{-score}}$	0.6650	0.7840	0.8575	0.7269	0.6339	0.7317	0.7022	0.6640	0.8586

(a)

(b)

FIGURE 6: Continued.

(c)

(d)

(e)

(f)
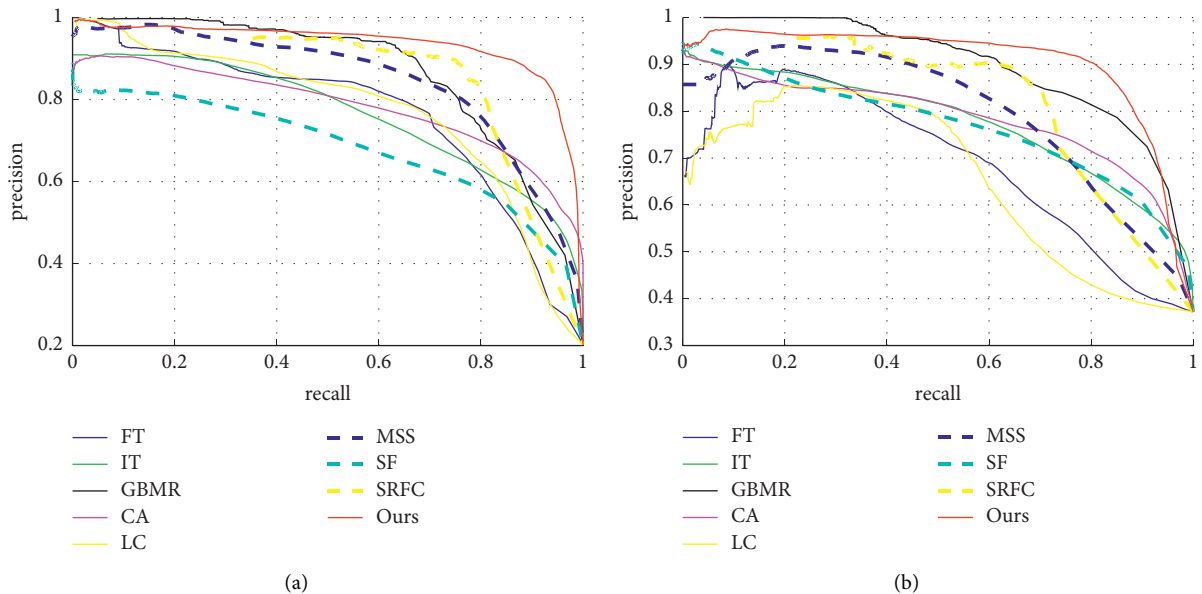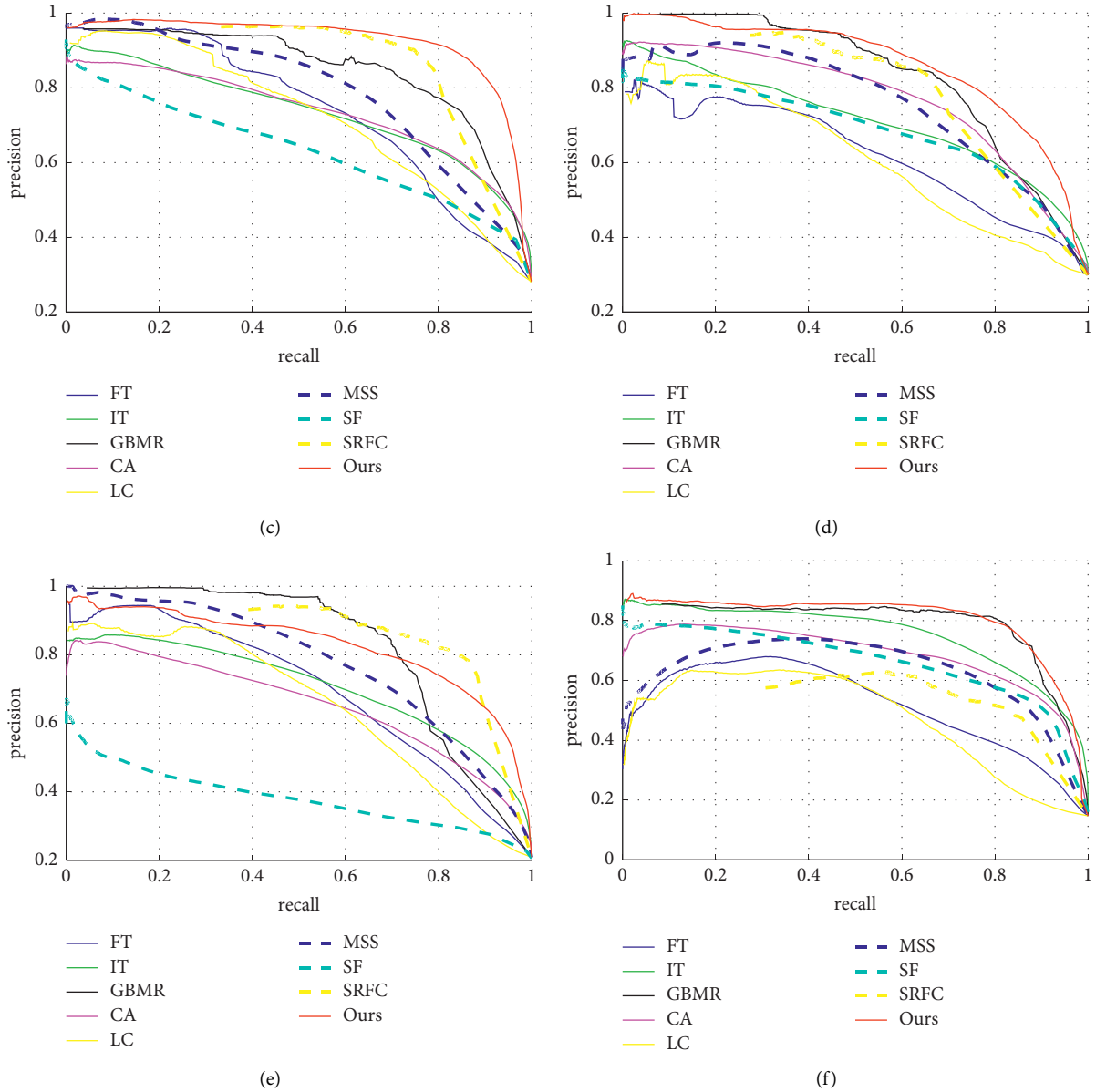
FIGURE 6: PR curves over different datasets. (a) MASK10K; (b) SOD; (c) DUTS; (d) ECSSD; (e) HKU_IS; and (f) DADA-W.

detection performance of the proposed method against the other methods. Table 1 shows a detailed comparison, in which the best three metrics are marked with red, blue, and green, respectively. From Table 1, we can see that the proposed method is always ranked in the top three over all datasets, which means that our method has certain versatility for various images. Compared to other methods, our method increases by an average of 23%, 11%, and 9% in terms of MAE scores, F_{β}-scores, and AUC scores, respectively.

Furthermore, the precision-recall (PR) curve and the receiver operating characteristics (ROC) curve are also used for intuitive comparison. Figures 6 and 7, respectively, show the PR and ROC curves of different algorithms over six datasets. It is clear that the proposed method achieves the most desirable detection performance. Concretely, the PR and ROC curves of our method are much better than those of FT, SF, and LC, but slightly better than those of IT, CA, and MSS. The reason is that these methods do not make full use of image background information. On the other hand, the PR and ROC curves of GBMR and SRFC are close to those of our method. However, when there exist multiple salient regions in the image or high color contrast in a salient region, our method clearly outperforms both of them.
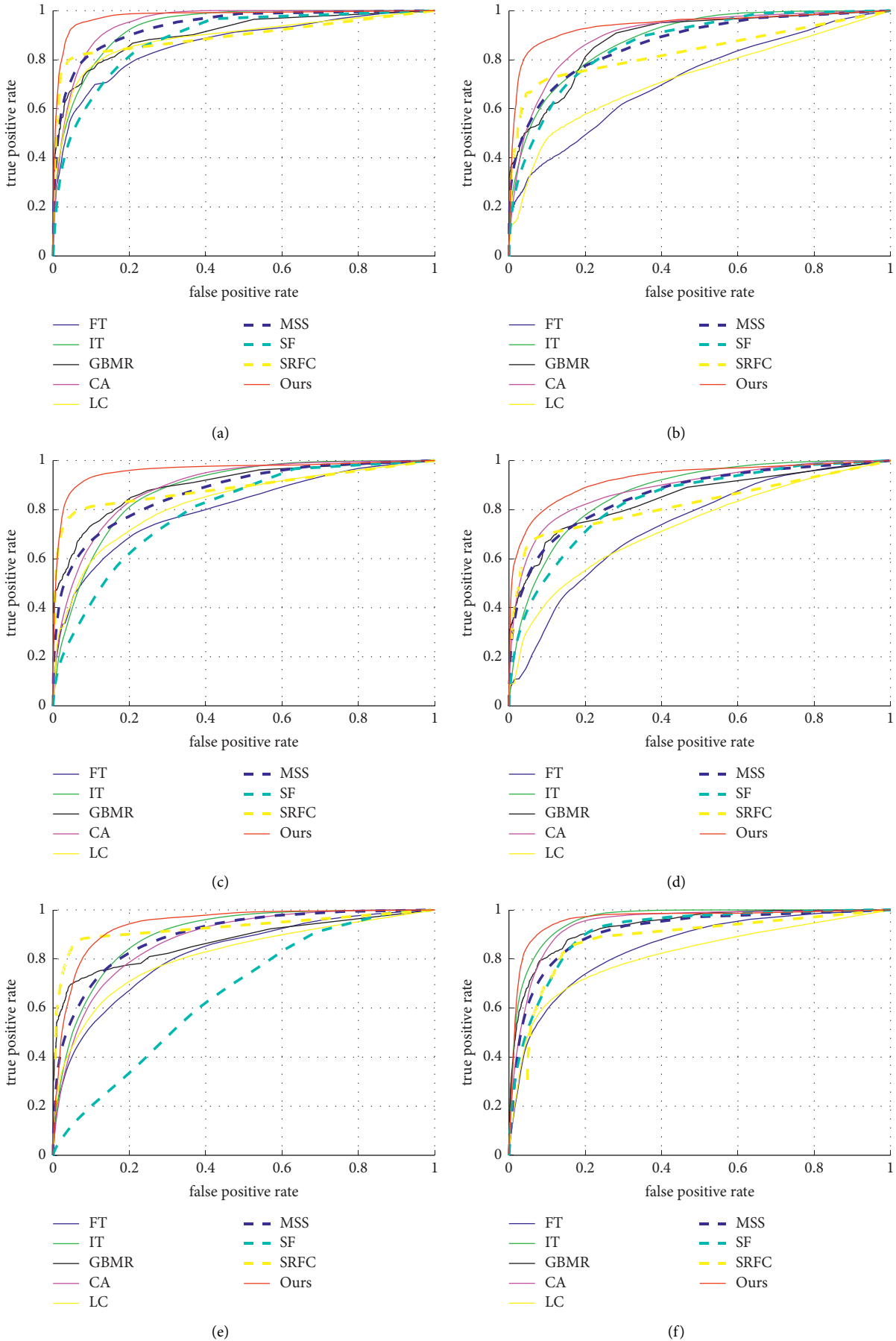
Figure 7: ROC curves over different datasets. (a) MASK10K; (b) SOD; (c) DUTS; (d) ECSSD; (e) HKU_IS; and (f) DADA-W.

4. Conclusion

In this study, we propose a novel and effective saliency detection algorithm. First, we create an accurate background template using background optimization selection. Based on this, saliency detection based on color contrast can be significantly improved to better highlight the salient region. Then, the saliency detection based on hash fingerprint is introduced to suppress the background. Finally, due to the two maps being complementary to each other, the final saliency map is obtained by fusing the two detection results. Experimental results show that the proposed algorithm achieves good detection results and outperforms other traditional saliency detection algorithms.

Data Availability

Five representative public datasets are available, as described in References, specifically, DUTS [24], ECSSD [25], MSRA10K [26], SOD [27], and HKU_IS [28].

Conflicts of Interest

The authors declare no conflicts of interest.

Authors' Contributions

Yong Wang contributed to conceptualization, original draft preparation, and funding acquisition; Yin Lv and Xuanrui Zhang were involved in formal analysis and validation.

Acknowledgments

References

[1] Z. Ren, S. Gao, L. T. Chia, and I. 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] U. Rutishauser, D. Walther, C. Koch, and P. Perona, "Is bottom-up attention useful for object recognition?" in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2004.

[3] C. Guo and L. Zhang, "A Novel Multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 185–198, 2010.

[4] L. Li, S. Jiang, Z. J. Zha, Z. Wu, and Q. Huang, "Partial-duplicate image retrieval via saliency-guided visual matching," *IEEE MultiMedia*, vol. 20, no. 3, pp. 13–23, 2013.

[5] K. Huang, C. Zhu, and G. Li, "Robust salient object detection via fusing foreground and background priors," in *Proceedings of the 25th IEEE International Conference on Image Processing*, pp. 2341–2345, IEEE, Athens, Greece, October 2018.

[6] Z. S. Wang and B. X. Li, "A two-stage approach to saliency detection in images," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 965–968, IEEE, Las Vegas, NV, USA, May 2008.

[7] T. Liu, J. Sun, N. Zheng, X. Tang, and H. Shum, "Learning to detect a salient object," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, Minneapolis, MN, USA, June 2007.

[8] L. Wang, J. R. Xue, N. N. Zheng, and G. Hua, "Automatic salient object extraction with contextual cue," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 105–112, IEEE, Barcelona, Spain, November 2011.

[9] S. Lu, V. Mahadevan, and N. Vasconcelos, "Learning optimal seeds for diffusion-based salient object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2790–2797, IEEE, Columbus, OH, USA, June 2014.

[10] J. F. Dai, Y. Li, K. He, J. Sun, and R. Fcn, "Object detection via region-based fully convolutional networks," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 379–387, IEEE, May 2016.

[11] 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 the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2083–2090, IEEE, Portland, OR, USA, Jyne 2013.

[12] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[13] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1597–1604, IEEE, Miami, FL, USA, June 2009.

[14] Y. Zhai and M. Shah, "Visual attention detection in video sequences using spatiotemporal cues," in *Proceedings of the 14th Annual ACM International Conference on Multimedia*, pp. 815–824, IEEE, July 2006.

[15] R. Achanta and S. Süsstrunk, "Saliency detection using maximum symmetric surround," in *Proceedings of the IEEE International Conference on Image Processing*, pp. 2653–2656, IEEE, Hong Kong, China, September 2010.

[16] Y. C. Wei, F. Wen, W. J. Zhu, and J. Sun, "Geodesic saliency using background priors," in *Proceedings of the European Conference on Computer Vision*, pp. 29–42, IEEE, Berlin, Heidelberg, July 2012.

[17] C. Yang, L. Zhang, H. Lu, X. Ruan, and M. Yang, "Saliency detection via graph-based manifold ranking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3166–3173, IEEE, Portland, OR, USA, June 2013.

[18] W. J. Zhu, S. Liang, Y. C. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2814–2821, IEEE, Columbus, OH, USA, June 2014.

[19] X. F. Zhang, Y. Wang, Z. X. Chen, J. Yan, and D. H. Wang, "Saliency detection via image sparse representation and color features combination," *Multimedia Tools and Applications*, vol. 79, no. 31-32, pp. 23147–23159, 2020.

[20] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[21] J. P. Wang, H. C. Lu, X. H. Li, N. Tong, and W. Liu, "Saliency detection via background and foreground seed selection," *Neurocomputing*, vol. 152, pp. 359–368, 2015.

[22] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 1915–1926, 2012.

[23] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: contrast based filtering for salient region detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 733–740, IEEE, Providence, RI, USA, June 2012.

[24] L. J. Wang, H. C. Lu, Y. F. Wang et al., "Learning to detect salient objects with image-level supervision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3796–3805, IEEE, Honolulu, HI, USA, July 2017.

[25] Q. Yan, L. Xu, J. P. Shi, and J. Y. Jia, "Hierarchical saliency detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1155–1162, IEEE, Portland, OR, USA, June 2013.

[26] M. M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S. M. Hu, "Global contrast based salient region detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 569–582, 2015.

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

[28] G. B. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proceedings Of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5455–5463, IEEE, 2015.