Journal of AI and Data Mining
Vol 6, No 1, 2018, 145-156

Graph-based Visual Saliency Model using Background Color

Sh. Foolad[1] and A. Maleki[2*]

1. Department of Electrical & Computer Engineering, Semnan University, Semnan, Iran.
2. Faculty of Biomedical Engineering, Semnan University, Semnan, Iran.

Received 03 July 2015; Revised 12 March 2016; Accepted 15 March 2017
**Corresponding author: amaleki@semnan.ac.ir (A. Maleki).*

Abstract
Visual saliency is a cognitive psychology concept that makes some stimuli of a scene stand out relative to their neighbors and attract our attention. Computing visual saliency is a topic of recent interest. Here, we propose a graph-based method for saliency detection, which contains three stages: pre-processing, initial saliency detection, and final saliency detection. The initial saliency map is obtained by putting an adaptive threshold on color differences relative to the background. In final saliency detection, a graph is constructed, and the ranking technique is exploited. In the proposed method, the background is suppressed effectively, and salient regions are often selected correctly. The experimental results on the MSRA-1000 database demonstrate excellent performance and low computational complexity in comparison with the state-of-the-art methods.

Keywords: *Visual Attention, Bottom-up Model, Saliency Detection, Graph Based, Background Color.*

1. Introduction

Salient regions in a scene immediately grab one's attention. Detecting these regions in the field of vision has a computational complexity that makes it difficult even for brains [1], let alone a computer. Selecting salient regions is possible through the visual attention mechanisms. The visual attention models have many applications including target detection (e.g. finding military vehicles in a plain [2]); cutting images automatically [3]; robotic vision actions [4]; finding the tumor on a mammogram [5]; video compression [6], and image retargeting [7, 8].

From the viewpoint of information processing and attention control, there are two executive approaches. One is the bottom-up or stimulus-driven process, [9-23], which is determined exogenously by the characteristics of the stimuli themselves. Therefore, the bottom-up models select locations that have different features relative to their surroundings. Another is the top-down or goal-directed process, [32-34] which endogenously depends on the observer's intentions and demands of the task as well. Some models [24, 28] use a combination of the bottom-up and top-down features. In this work, we focused on the bottom-up saliency detection. The first attempt to understand the bottom-up attention was made by Koch and Ullman [9] (1985), whereas the first actual implementation of the corresponding saliency map was described by Niebur and Koch [10] (1996). Later works refined this model [11-14]. Itti et al. [11] proposed a bottom-up saliency detection method using center-surround differences. Their method detects the contour of the salient object instead of the whole object. Some methods search for saliency cues based on the frequency domain analysis [7, 8, 15-17]. Hou and Zhang [15] (2007) used global contrast to obtain the saliency map. In their model, edges are detected as the salient regions. Achanta et al. [16] (2009) proposed a method by maintaining the higher-frequency content of images. However, this method often recognizes some background regions as saliency. Recently, Fang et al. [7] (2012) has provided a method in the compressed domain, and calculated the saliency map by combining the feature maps (intensity, color, and texture) obtained from DCT coefficients. Also, Fang et al. [8] (2012) have presented a bottom-up method by calculating differences of a quaternion Fourier transform (QFT) between each patch and the other patches in the image. They exploited

human visual sensitivity to weight these patch differences. Some other bottom-up methods define visual saliency based on the graphical framework [18- 20, 30]. Harel et al. [18] (2006) formed some activation maps using the principles of closeness and dissimilarity and combined them into a single map. These models highlight only the high-contrast edges and do not detect the interior smooth objects. Most saliency detection models use center prior [21, 27, 31], assuming that the objects near the image center are more prominent than the other regions. These models cannot detect salient regions far from the image center. Yang [20] solved the problem by estimating the center of the salient object based on the convex hull technique. However, the performance of the method depends on the accuracy of the technique for salient area extraction.

Most methods use the color, orientation, intensity, and frequency-based features for directing attention. Ma and Zhang [22] (2003) estimated saliency using a fuzzy growing method and considering only local contrasts. Hence, this method detects the object boundaries regardless of the inside of the salient regions. Achanta et al. [23] (2008) presented a bottom-up model using the color and luminance features. Borji [24] (2012) applied low-level features such as intensity, orientation and color, and high-level features such as faces, humans, and cars for saliency detection. Zhang et al. [25] (2013) proposed a saliency detection model by combining frequency, color, and location priors. Erdem [26] (2013) used the covariance of the features extracted from the image patches. Then the distances between the covariance of each patch with neighboring patches were computed for saliency estimation. Tian et al. [28] (2014) proposed a model by fusing the bottom-up (color and orientation contrast) and top-down (depth-from-focus) features.

The above-mentioned methods cannot correctly recognize the salient regions in most scenes. Determining the location of salient pixels without knowing their characteristics is difficult. In this paper, we propose a bottom-up model that consists of three stages: First, pre-processing of the input image is performed that includes segmentation into superpixels, conversion from RGB to LAB space, and calculating color mean for each LAB channel of each superpixel. Then an initial saliency map is achieved based on an adaptive threshold of the color differences relative to the background. In the end, the final saliency map is obtained by graph construction and using the ranking technique [29, 30]. In this method, the background is suppressed effectively and the salient object is highlighted more accurately in the image.

The rest of the paper is organized as follows. The proposed method is described in section 2. The experimental results on the popular large database are presented in section 3. The conclusion is discussed in section 4.

2. Proposed method

We propose a graph-based method using the background color for saliency estimation. As illustrated in figure 1, our method consists of three stages. In the first stage, we perform pre-processing on the input image. In the second stage, we detect the background and foreground regions based on the color differences in the LAB space. The color is a very important visual attribute, and the LAB space is inspired by color opponent neurons of red/green and blue/yellow in primary visual cortex (V1). Usually, the boundary regions of an image are considered as background, thus we also assume these regions as part of the background. The rest of the background and foreground regions are determined adaptively based on the color differences relative to the already assumed background regions. Accordingly, an initial saliency map is generated. In the third stage, the ranking technique [29, 30] is exploited by taking the initial salient regions as queries. Final saliency is computed based on the relevance of queries.

2.1. Pre-processing

Pre-processing is performed through three procedures. First, a given image is segmented into N superpixels using the simple linear iterative clustering (SLIC) algorithm [35]. The SLIC algorithm follows four main steps: 1) N cluster centers, $C_n = [l_n, a_n, b_n, x_n, y_n]^T$, are initialized in a regular grid space S using k-means algorithm. 2) In a $2S \times 2S$ region around each cluster center, the distance between the cluster center and each one of the pixels within the region is calculated in terms of color similarity and spatial proximity. 3) Each pixel is then associated with the nearest cluster center. Then the color average of all pixels belonging to each cluster center is computed and the color of the cluster center is updated. 4) The error is calculated between the previous and new cluster centers, and steps 2 to 4 can be repeated until the error converges. At the end of the algorithm, each cluster center, along with the associated pixels, is called a superpixel.

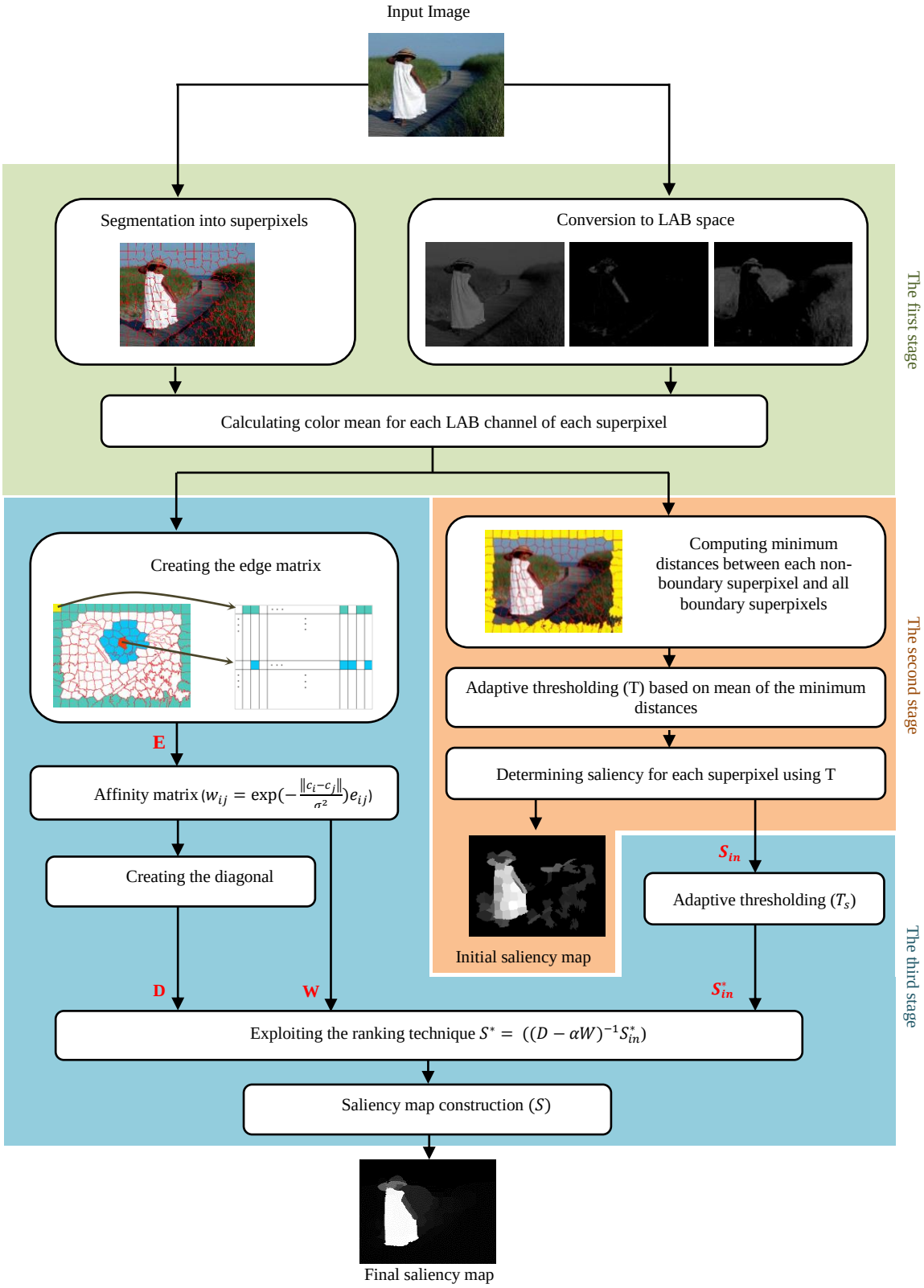


Figure 1. Diagram of proposed method. This method consists of three stages: pre-processing, initial saliency detection, and final saliency detection.

Therefore, a superpixel contains the pixels that are more uniform in terms of color and texture. Using superpixels, the structure of the salient objects is conserved, whereas the computational complexity of the image is efficiently decreased from thousands of pixels to only a few hundred superpixels. Furthermore, the superpixels maintain the image edges well and improve the

quality of salient object detection. Figure 2 illustrates some images that are segmented into superpixels using the SLIC algorithm.

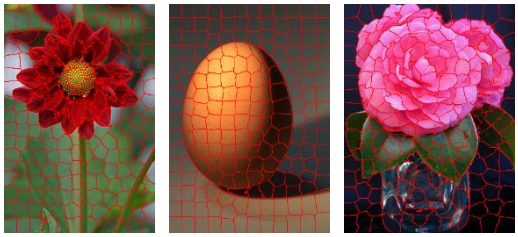


Figure 2. Images segmented using SLIC algorithm into about 200 superpixels.

In the second procedure of pre-processing, the given image is migrated from RGB to LAB space. Since the LAB color space is uniform and similar to human visual perception [36], it is a good choice for computing the color differences. LAB contains a luminance (L) channel and two chromatic channels: red/green (A) and blue/yellow (B). In the LAB space, the L, A, and B channels are computed through a non-linear mapping of XYZ coordinates, as follows [37]:

$$L = 116\, h\left(\frac{Y}{Y_w}\right) - 16 \tag{1}$$

$$\mathrm{A} = 500\left[h\left(\frac{X}{X_w}\right) - h\left(\frac{Y}{Y_w}\right)\right] \tag{2}$$

$$\mathrm{B} = 200\left[h\left(\frac{Y}{Y_w}\right) - h\left(\frac{Z}{Z_w}\right)\right] \tag{3}$$

$$h(q) = \begin{cases} \sqrt[3]{q} & q > 0.008856 \\ 7.787q + \dfrac{16}{116} & q \le 0.008856 \end{cases} \tag{4}$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{5}$$

where, X_w, Y_w, and Z_w are reference white points, and their values are 0.950450, 1.000000, and 1.088754, respectively.

In the third procedure of pre-processing, the color mean is computed for each LAB channel of the ith superpixel, as follows:

$$l^*_i = \frac{1}{M_i}\sum_{m=1}^{M_i} L(m) \tag{6}$$

$$a^*_i = \frac{1}{M_i}\sum_{m=1}^{M_i} A(m) \tag{7}$$

$$b^*_i = \frac{1}{M_i}\sum_{m=1}^{M_i} B(m) \tag{8}$$

where, L, A, and B are three channels in the LAB space and M_i is the number of pixels within the ith superpixel.

2.2. Initial saliency detection

Similar to our previous work [38], we detected the background and foreground superpixels by color distances. The distance between two superpixels (superpixels i and j) was computed as follows:

$$D(i.j) = c_i - c_j = \sqrt{(l^*_i - l^*_j)^2 + (a^*_i - a^*_j)^2 + (b^*_i - b^*_j)^2} \tag{9}$$

where, $c_i = \left[l^*_i, a^*_i, b^*_i\right]$ and $c_j = \left[l^*_j, a^*_j, b^*_j\right]$ denote color means of the pixels within the superpixels i and j, respectively, and $\|.\|$ is the Euclidean distance.

The superpixels are divided into two groups: boundary and non-boundary. "Boundary superpixels" refer to the superpixels of the image boundary, and the rest are "non-boundary superpixels" (see Figure 3). Given N superpixels in the image, the R superpixels are boundary superpixels and the K superpixels are non-boundary superpixels (N = R + K). All the boundary superpixels are assumed as background, and the non-boundary superpixels have to be evaluated whether as background or foreground. To do this, the distances between each non-boundary superpixel and all boundary superpixels are computed as (9), and the minimum distance is achieved as follows:

$$MD(i) = \min_j D(i,j) \qquad j = 1,2,\ldots,R \tag{10}$$

where, i characterizes the non-boundary superpixel and $MD(i)$ is the minimum distance between it and all boundary superpixels $(j = 1,2,\ldots,R)$. If the distances between the non-boundary superpixel and all boundary superpixels are high, its minimum distance would be high as well. Therefore, it is not similar to the boundary superpixels or background, and would be considered as a foreground superpixel. The mean of minimum distance is considered as an adaptive threshold (T).

$$T = \frac{1}{K}\sum_{i=1}^{K} MD(i) \tag{11}$$

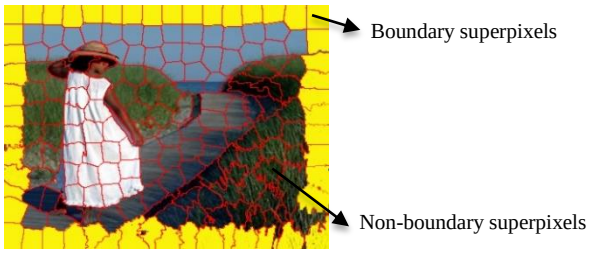


Figure 3. Boundary and non-boundary superpixels in a sample image. Yellow superpixels are boundary and others are non-boundary.

Each non-boundary superpixel whose minimum distance is bigger than T is defined as foreground. The initial saliency value for each superpixel is determined as follows:

$$S_{in}(n) = \begin{cases} MD(n) & if\ MD(n) > T\ and \\ & n \in \{K\ \ non-boundary\ \ superpixels\} \\ 0 & if\ MD(n) \le T\ and \\ & n \in \{K\ \ non-boundary\ \ superpixels\} \\ 0 & if\ n \in \{R\ \ boundary\ \ superpixels\} \end{cases} \quad (12)$$

where, $S_{in}(n)$ is the initial saliency value for superpixel n. For creating the initial saliency map, we set the initial saliency value of each superpixel for all pixels within it. For instance, the initial saliency maps of some images with their ground truths are shown in figure 4.

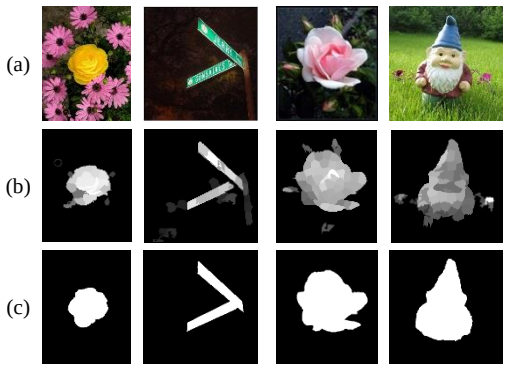


Figure 4. Examples of initial saliency maps. (a) Input images, (b) Initial saliency maps, (c) Ground truths.

Sometimes the background color is similar to the salient object color in the image. Therefore, the initial saliency map cannot recognize the salient object well, and parts of the background regions are detected as foreground incorrectly. Examples of the initial saliency maps in figure 4(b) illustrate the problem. To alleviate the problem, we used the ranking technique based on graph construction.

2.3. Final saliency detection

We represent the image as a graph with superpixels as nodes (explained in section 2.3.1) and the initial salient nodes as queries. Then we exploit the ranking technique (described in section 2.3.2). The saliency of each node is obtained by its relevance to queries, so the final saliency map is created as the result.

2.3.1 Graph construction

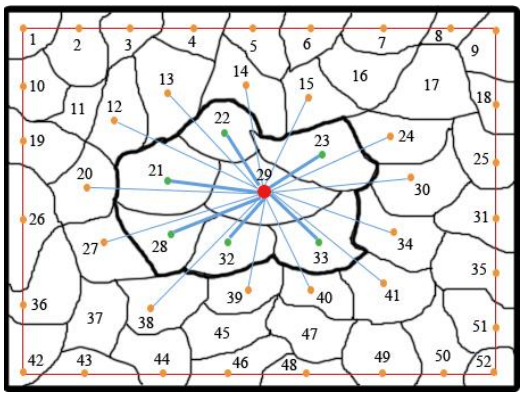
We construct a connected graph G = (V, E), as shown in figure 5(a), where the nodes (V) are superpixels generated by the SLIC algorithm [35]. The edges (E) are determined by the relationship between superpixels; if two superpixels have a frontier, there is an edge between them. For example, in figure 5(a), superpixel 29 has frontiers with superpixels 21, 22, 23, 33, 32, and 28. Thus there is an edge between superpixel 29 and each of them. Each node is connected to its neighboring nodes (See the bold blue lines in figure 5(a)) and those that have frontiers with the neighboring nodes (See the blue lines in figure 5(a)). Furthermore, all the nodes on the four sides of the image are connected together (See the red line in figure 5(a)). An $N \times N$ edge matrix (E) is generated, in which $e_{ij} = 1$. If there is an edge between nodes i and j, otherwise $e_{ij} = 0$. An example of the edge matrix is illustrated in figure 5(b). According to [20, 30], the weight between the two nodes i and j is computed as follows:

$$w_{ij} = \exp\left(-\frac{D(i,j)}{\sigma^2}\right)e_{ij} = \exp\left(-\frac{c_i - c_j}{\sigma^2}\right)e_{ij} \quad (13)$$

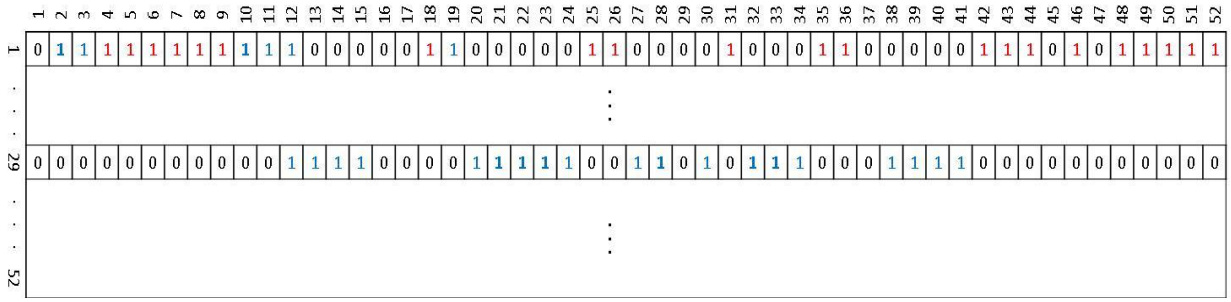
where, $D(i,j)$ is computed as (9), e_{ij} is the element (i,j) of the edge matrix (E), and σ controls the strength of the weight. In (13), the affinity matrix (W) is created by determining distances of the nodes in the LAB color space. The weights are increased by reducing the color differences between nodes.

2.3.2 Ranking technique

To achieve the final saliency map, we exploit the ranking technique [29]. First, we apply an adaptive threshold on S_{in} in (12) and detect the initial salient nodes.



(a)



(b)

Figure 5. (a) A sample of our graph model. Blue lines indicate edges of red node (node 29), and red line shows that all boundary nodes are connected together. (b) Parts of 52×52 edge matrix (E) for graph (a) is shown edges of nodes 1 and 29. For each node in E, bold blue 1's indicate its edges with neighboring nodes; blue 1's refer to its edges with neighboring of them. If a node is boundary (e.g. node 1), red 1's represents its edges with boundary nodes.

$$T_s = \frac{1}{N}\sum_{n=1}^{N} S_{in}(n) \tag{14}$$

$$S_{in}^{*}(n) = \begin{cases} 1 & if\ S_{in}(n) \geq T_s \\ 0 & if\ S_{in}(n) < T_s \end{cases} \tag{15}$$

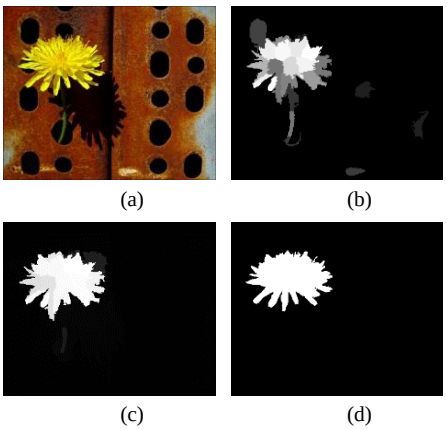
The nodes with $S_{in}^{*} = 1.$ are defined as initial salient nodes. We consider the initial salient nodes as queries for graph labelling.

The saliencies of the unlabelled nodes are computed based on their relevance to queries by the ranking technique:

$$S^{*} = (D - \alpha W)^{-1} S_{in}^{*} \tag{16}$$

where, W is the affinity matrix computed in (13), D denotes the diagonal matrix with $d_{ii} = \sum_{j}(w_{ij})$. and α is the impact factor of the affinity matrix in the ranking. In (16), after computing $(D - \alpha W)^{-1}$, we set its diagonal elements to zero. This prevents the relevance of each query to itself, and weakens the contributions of other queries to the ranking [30]. Therefore, it has good effects on the results.

To generate the final saliency map (S), we set the saliency value of each node for all pixels within its corresponding superpixel, and normalize it to the range [0 1]. Figure 6 shows the impact of the graph-based ranking technique on the final saliency map.



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

Figure 6. (a) Input image, (b) Initial saliency map, (c) Final saliency map, (d) Ground truth.

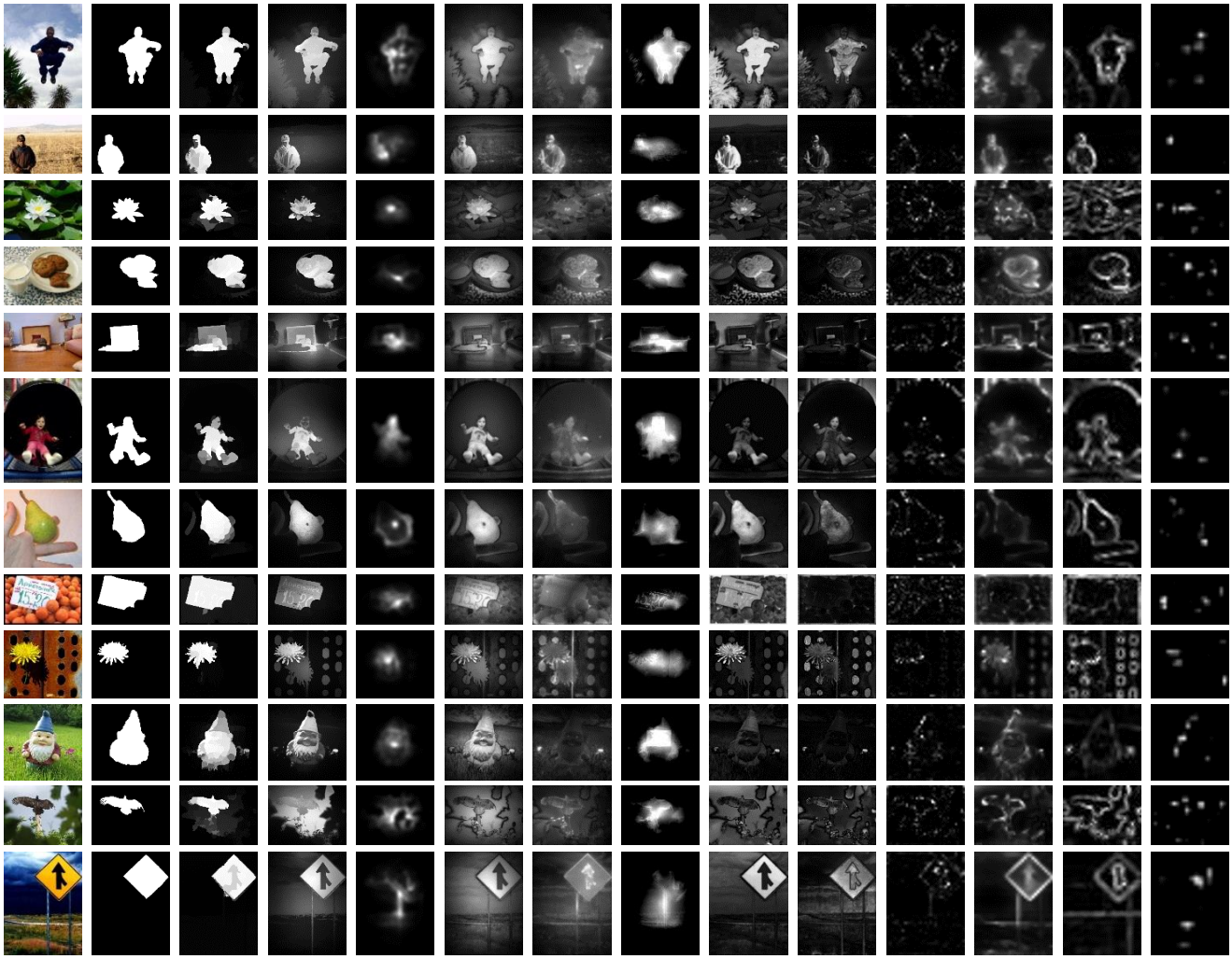


Figure 7. Visual comparison of saliency maps obtained from state-of-the-art methods and our method (ours) with ground truth (GT).

3. Experimental results

We demonstrate the efficacy of the proposed method in a series of experiments, while comparing it with the state-of-the-art saliency detection models: Itti et al. [11] (IT), Ma and Zhang [22] (MZ), graph-based visual saliency [18] (GB), spectral residual approach [15] (SR), Achanta [23] (AC), frequency-tuned saliency detection [16] (FT), a model in compressed domain [7] (CD), boosting bottom-up and top-down model [24] (BT), a model based on wavelet transform [27] (WV), a saliency detection model that combines simple priors [25] (SDSP), visual saliency using region covariance [26] (CS), and a model that combines the orientation, color, and depth-from-focus features [28] (FF). In the paper, we used the original experiment results of models FF, BT, FT, AC, SR, GB, MZ,nd IT that are available online[1], and downloaded the results of model CD from the authors' homepage[2]. Furthermore, we retrieved the saliency maps of other models including CS, SDSP, and WV from their source codes[3]. Visual comparison of the state-of-the-art methods and the proposed method with the ground truth are shown in figure 7. As shown, the proposed method is closer to ground truth than the other methods are.

3.1. Database

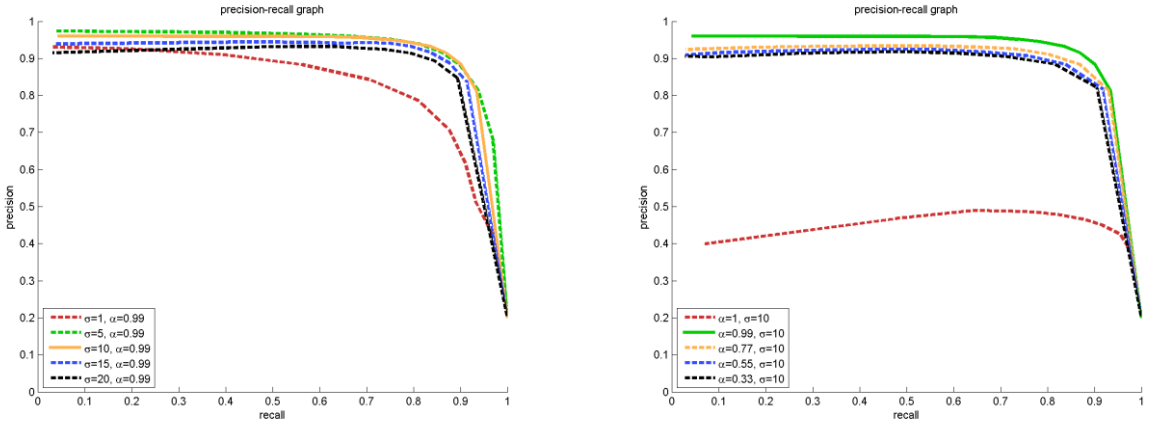
In our experiment, we used the popular database of MSRA-1000, a subset of the MSRA database

[1]https://sites.google.com/site/tianhuawei/project/saliencydetection

[2]https://sites.google.com/site/leofangyuming/Home/smap-compressed-domain

[3]http://web.cs.hacettepe.edu.tr/~erkut/projects/CovSal,http://sse.tongji.edu.cn/linzhang/va/SDSP/SDSP.htm, https://sites.google.com/site/leofangyuming/Home

[39], which contains 1000 images with the binary ground truths. The Ground truth was determined by the visual judgment of 9 subjects.



(a) (b)

Figure 8. Comparison of precision-recall curves of proposed method on MSRA-1000 database for different values of (a) σ . parameter and (b) α parameter.

3.2. Parameters

In our experiments, we set the number of superpixels to $N = 200$. There are also two other parameters in our implementation, σ and α. These are used for controlling the edge weight in (13) and the impact of the affinity matrix in the ranking function in (16), respectively, and were empirically tuned as $\sigma = 10$ and $\alpha = 0.99$. Precision-recall curves, which determine the effects of different parameters on the MSRA-1000 database, are shown in figure 8.

3.3. Evaluation criteria

We evaluated the performance of the methods by receiver operating characteristic (ROC), precision, recall, and F-measure. In this experiment, the saliency map values in the range [0 255] are converted to binary by a fixed threshold. First, the values for the saliency map are normalized to the range [0 1]. Then given a threshold, $T_f \in [0\ 1]$, the points with saliency values higher than T_f are marked as salient and others marked as non-salient. For each image, we generate several binary saliency maps corresponding to 11 different thresholds, from 0 to 1: (0, 0.1, 0.2, …, 1). This operation is applied to all saliency maps obtained from the database images. For each image, the binary saliency maps obtained from each saliency detection model are compared with its ground truth map. Afterward, true positive, false positive, and false negative are computed, an example of which is shown in figure 9.

(a) (b) (c)

Figure 9. Classification results (a) with ground truth map (b) and a binary saliency map with a threshold equal 0.6 (c). In (a), green areas are true positives, blue areas are false positives, and red areas are false negatives.

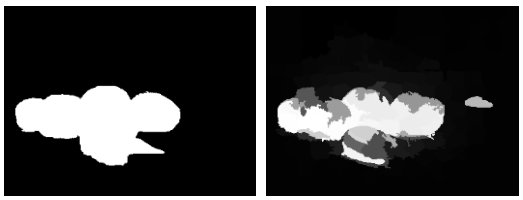
True positive is the number of pixels that are salient in both the binary saliency map and the ground truth map. False positive is the number of pixels that are salient in binary saliency map and are non-salient in the ground truth map. False negative is the number of pixels that are salient in the ground truth map and are non-salient in the binary saliency map. The precision and recall rates for each binary saliency map are calculated as follows [28]:

$$precision = \frac{TP}{TP + FP} \tag{17}$$

$$recall = \frac{TP}{TP + FN} \tag{18}$$

where, TP, FP, and FN are the true positive, false positive, and false negative rates, respectively. The precision value is the ratio of correctly detected salient pixels to all salient pixels of binary saliency map, while the recall value is the ratio of correctly detected salient pixels to all salient pixels of the ground truth.

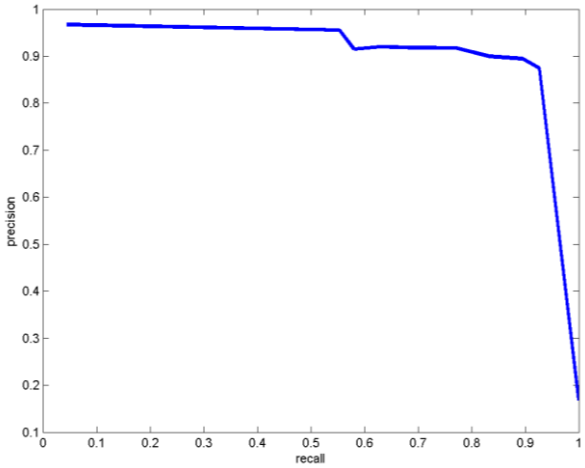
Given a ground truth map SM_G and a predicted map SM_p :



SM_G SM_p

Define a set of thresholds $T = \{T_p^i\}_{i \in N}$
For All thresholds in T
  Binarization of the predicted map with the threshold T_p^i.
  For All pixels in binarized maps
    If $SM_G = 1$
      If $SM_p^i = 1$ // SM_p^i *is binarized map with threshold* T_p^i
        $TP{+}{+}$
      Else
        $FN{+}{+}$
      End If
    Else
      If $SM_p^i = 1$
        $FP{+}{+}$
      End If
    End If
  End For
  Precision(T_p^i)=TP/ (TP+FP)
  Recall(T_p^i)=TP/ (TP+FN)
End For
Plot (Recall, Precision) //*for each* T_p^i

(a) Pseudo code



(b) presicion-recall curve

Figure 10. (a) Pseudo-code to perform a precision-recall analysis between ground truth map and predicted map. (b) the precision-recall curve for a sample predicted map.

The values for precision and recall are obtained from the binary saliency maps of all the database images, and the precision-recall curve is drawn by averaging the results for each threshold value T_f. Pseudo-code for drawing a precision-recall curve is shown in figure 10. By the precision and recall criteria, the F-measure is obtained. Similar to [16], for each image, its gray saliency map is converted into binary by adaptive threshold (Th) that is two times the mean saliency from the saliency map.

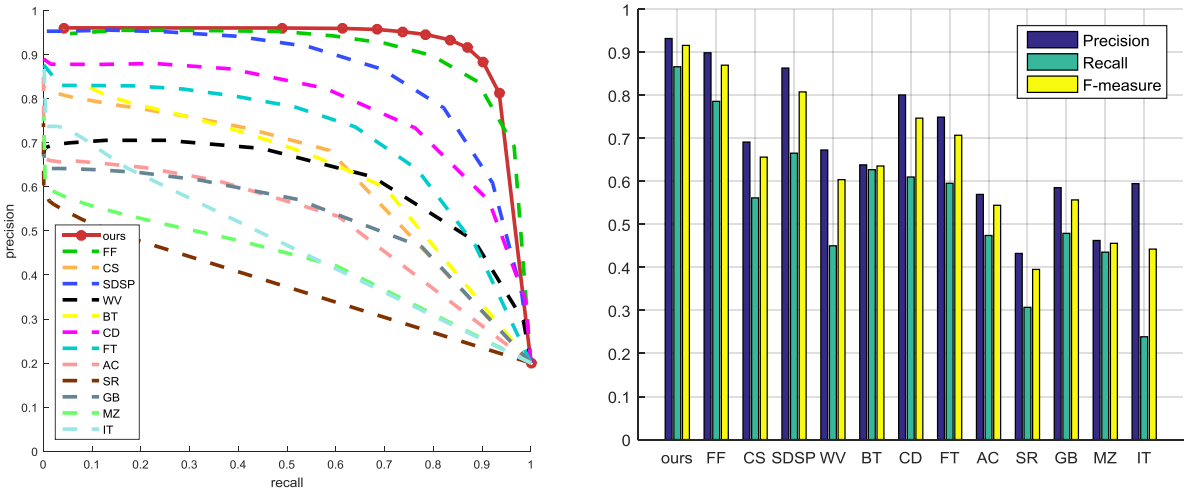
$$Th = \frac{2}{X * Y} \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} S(x, y) \qquad (19)$$

where, X and Y are the width and height of the saliency map, respectively, and $S(x, y)$ is the saliency value of the pixel at the position (x, y).

The precision and recall are computed for each image. Then by averaging them on all the images, the F-measure is defined as:

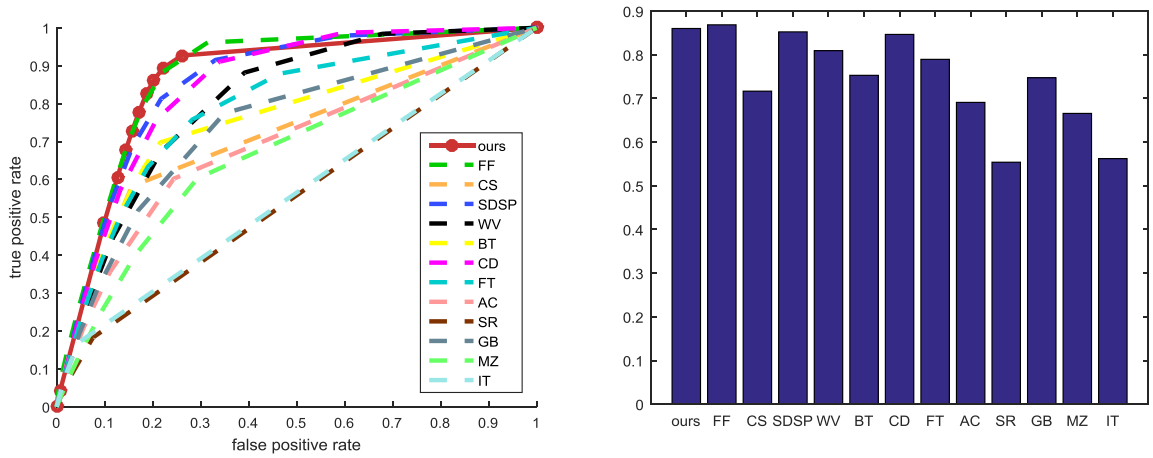
$$F - measure = \frac{(1 + \beta^2).\, precision.recall}{\beta^2.\ precision + \ recall} \qquad (20)$$

where, $\beta^2 = 0.3$ as in [16, 25, 31] for more impact of precision than recall. The F-measure criterion for the state-of-the-art methods on MSRA-1000 database is listed in table 1.
Another way to compare experiments is to use the ROC curve that is achieved as the curve of true positive rate versus false positive rate by different thresholds over the saliency map of each image. The area under the ROC curve (AUC) [40] can be used for the quantitative assessment of the saliency detection models.



(a) (b)

Figure 11. (a) Precision-recall curves of different methods. (b) Precision, recall and F-measure using an adaptive threshold. All results are computed on MSRA-1000 database for comparison of proposed method (ours) with state-of-the-art methods including FF [28], CS [26], SDSP [25], WV [27], BT [24], CD [7], FT [16], AC [23], SR [15], GB [18], MZ [22], IT [11].



(a) (b)

Figure 12. ROC curves (a) and AUC scores (b) for saliency models (ours is the proposed method).

Table 1. F-measure for saliency models.

Method	F-Measure
ours	**0.9093**
FF [28]	0.8701
CS [26]	0.6564
SDSP [25]	0.8080
WV [27]	0.6040
BT [24]	0.6358
CD [7]	0.7470
FT [16]	0.7072
AC [23]	0.5444
SR [15]	0.3954
GB [18]	0.5569
MZ [22]	0.4561
IT [11]	0.4426

Table 2. Comparison of average running time.

Method	Time (s)	Code
ours	**1.0972**	**Matlab**
CS [26]	21.223	Matlab
SDSP [25]	0.087	Matlab
WV [27]	6.777	Matlab
FT [16]	0.024	C++
AC [23]	0.127	C++
SR [15]	0.061	Matlab
GB [18]	1.563	Matlab
MZ [22]	0.07	C++
IT [11]	0.411	Matlab

We used an available code[4] to achieve the ROC curve and AUC score.

For evaluating the performance of the saliency detection models, we compare the proposed method with twelve state-of-the-art methods including IT [11], MZ [22], GB [18], SR [15], AC [23], FT [16], CD [7], BT [24], WV [27], SDSP [25], CS [26], and FF [28] on the MSRA-1000 database [39]. The precision-recall curve and F-measure metric are shown in figure 11. In addition, we use the ROC curve and AUC score for the methods that are shown in figure 12. The results obtained show that our method outperforms the state-of-the-art saliency detection methods. In addition to the saliency detection accuracy, we compared the computational cost of the proposed method with other methods. The average running time for the saliency detection methods on the MSRA-1000 database is presented in table 2 on a machine with Intel Core i5 2.53 GHz CPU and 4 GB RAM. Our method has a quite low computational complexity.

4. Conclusions

We proposed a bottom-up visual saliency detection method that detects the background and foreground regions by the color differences relative to the boundary regions of the image, based on which an initial saliency map is generated. Next, with graph construction, the ranking technique is exploited by taking the initial salient nodes as queries. The final saliency map is created by this technique. In the proposed method, the background is suppressed effectively, and the whole salient regions are highlighted uniformly. Furthermore, due to using superpixels instead of pixels, the computational complexity of the method is effectively reduced. As indicated in the experimental results on the MSRA-1000 database, the proposed method demonstrates an excellent performance against the state-of-the-arts methods. Furthermore, having a quite low run time, this method can be used for real-time applications. Since the proposed method uses the color of the boundary regions as the background features, if these features are similar to salient regions, or the image has a frame, the salient regions are not detected correctly. Our future work will focus on generating a top-down method and its combination with the proposed bottom-up method, similar to the human visual system.

[4]https://github.com/cvzoya/saliency/blob/master/code_forMetrics/AUC_Borji.m

References

[1] Tsotsos, J. K. (1991). Is Complexity Theory appropriate for analyzing biological systems? Behavioral and Brain Sciences, vol. 14, no. 4, pp. 770-773.

[2] Itti, L. & Koch, C. (2000). A saliency-based search mechanism for overt and covert shifts of visual attention. Vision Research. vol. 40, no. 10, pp. 1489-1506.

[3] Le Meur, O., Le Callet, P., Barba, D. & Thoreau, D. (2006). A coherent computational approach to model bottom-up visual attention, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 5, pp. 802-817.

[4] Siagian, C. & Itti, L. (2007). Biologically-Inspired Robotics Vision Monte-Carlo Localization in the Outdoor Environment, In Proc. IEEE International Conference on Intelligent Robots and Systems.

[5] Hong, B. W. & Brady, M. (2003). A Topographic Representation for Mammogram Segmentation, In Lecture Notes in Computer Science, pp. 730-737.

[6] Itti, L. (2004). Automatic Foveation for Video Compression Using a Neurobiological Model of Visual Attention. IEEE Transactions on Image Processing, Vol. 13, no. 10, pp. 1304-1318.

[7] Fang, Y., Chen, Z., Lin, W. & Lin, C. W. (2012). Saliency detection in the compressed domain for adaptive image retargeting, IEEE Transactions on Image Processing, vol. 21, no. 9, pp. 3888-3901.

[8] Fang, Y., Lin, W., Lee, B. S., Lau, C. T., Chen, Z., & Lin, C. W. (2012). Bottom-up saliency detection model based on human visual sensitivity and amplitude spectrum, IEEE Transactions on Multimedia, vol. 12, no. 1, pp. 187-198.

[9] Koch, C. & Ullman, S. (1985). Shifts in selective visual attention: towards the underlying neural circuitry. Human Neurobiology 4, pp. 219-227.

[10] Niebur, E. & Koch, C. (1996). Control of Selective Visual Attention: Modeling the Where Pathway. Neural Information Processing Systems, pp. 802-808.

[11] Itti, L., Koch, C. & Niebur, E. (1998). 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.

[12] Itti, L. & Koch, C. (2001), Computational Modeling of Visual Attention, Nature Reviews Neuroscience, vol. 2, no. 3, pp. 194-203.

[13] Itti, L., Dhavale, N. & Pighin, F. (2004). Realistic Avatar Eye and Head Animation Using a Neurobiological Model of Visual Attention. Proc. SPIE, vol. 5200, no. 1, pp. 64-78.

[14] Lin, Y., Tang, Y., Fang, B., Shang, Z., Huang, Y., & Wang, S. (2013). A Visual-Attention Model Using Earth Mover's Distance-Based Saliency Measurement and Nonlinear Feature Combination. IEEE

Transactions on pattern analysis and machine intelligence, vol. 35, no. 2, pp. 314-328.

[15] Hou, X. & Zhang, L. (2007). Saliency Detection: A Spectral Residual Approach, Proc. IEEE Conference on Computer Vision and Pattern Recognition.

[16] Achanta, R., Hemami, S., Estrada, F. & Susstrunk, S. (2009). Frequency-tuned salient region detection. In Computer Vision and Pattern Recognition (CVPR), IEEE Conference, pp. 1597–1604.

[17] Guo, C. & Zhang, L. (2010). A Novel Multiresolution Spatiotemporal Saliency Detection Model and Its Applications in Image and Video Compression. IEEE Transaction on Image Processing, vol. 19, no. 1, pp. 185-198.

[18] Harel, J., Koch, C. & Perona, P. (2006). Graph-based visual saliency. Neural Information Processing Systems (NIPS), pp. 545–552.

[19] Avraham, T. & Lindenbaum, M. (2010). Esaliency (Extended Saliency): Meaningful Attention Using Stochastic Image Modeling. IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 32, no.4, pp. 693-708.

[20] Yang, C., Zhang, L. & Lu, H. (2013). Graph-Regularized Saliency Detection with Convex-Hull-Based Center Prior. IEEE Signal Processing Letters, vol. 20, no. 7, pp. 637-640.

[21] Goferman, S., Zelnik-Manor, L. & Tal, A. (2012). Context-aware saliency detection, IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 32, no. 10, pp. 1915-1925.

[22] Ma, Y. & Zhang, H. (2003). Contrast-based image attention analysis by using fuzzy growing. In International Multimedia Conference: Proceedings of the eleventh ACM international conference on Multimedia, vol. 2, no. 8, pp. 374-381.

[23] Achanta, R., Estrada, F., Wils, P. & Susstrunk, S. (2008). Salient region detection and segmentation. International Conference on Computer Vision Systems (ICVS), pp. 66-75.

[24] Borji, A. (2012). Boosting bottom-up and top-down visual features for saliency estimation. IEEE Conference on Computer Vision Pattern Recognition (CVPR), pp. 438-445.

[25] Zhang, L., Gu, Z. & Li, H. (2013). SDSP: A Novel Saliency Detection Method by Combining Simple Priors, in Proc. ICIP, pp. 171-175.

[26] Erdem, E. & Erdem, A. (2013). Visual saliency estimation by nonlinearly integrating features using region covariances. Journal of Vision, vol. 13, no. 4, pp. 1-20.

[27] İmamoğlu, N., Lin, W. & Fang, Y. (2013). A Saliency Detection Model Using Low-Level Features Based on Wavelet Transform. IEEE Transactions on Multimedia, vol. 15, no. 1, pp. 96-105.

[28] Tian, H., Fang, Y., Zhao, Y., Lin, W., Ni R. & Zhu, Z. (2014). Salient region detection by fusing bottom-up and top-down features extracted from a single image. IEEE Transactions on image processing, vol. 23, no. 10, pp. 4389-4398.

[29] Zhou, D., Weston, J., Gretton, A., Bousquet, O. & Scholkopf, B. (2004). Ranking on data manifolds. In NIPS.

[30] Yang, C., Zhang, L., Lu, H., Ruan, X. & Yang, M. H. (2013). Saliency Detection via Graph-based Manifold Ranking. IEEE Conference on Computer Vision and Pattern Recognition, Portland, pp. 1-8.

[31] Zhou, Q. (2014). Object-based attention: saliency detection using contrast via background prototypes, electronics letters, vol. 50, no. 14, pp. 997-999.

[32] Borji, A., Sihite, D. N. & Itti, L. (2014). What/Where to Look Next? Modeling Top-Down Visual Attention in Complex Interactive Environments. IEEE Transactions on systems man and Cybernetics systems, vol. 44, pp. 523-538.

[33] Karthikeyan, S., Jagadeesh, V. & Manjunath, B. S. (2013). Learning top-down scene context for visual attention modeling in natural images, 20th IEEE international conference on image processing (ICIP), pp. 211-215.

[34] Borji, A., Sihite, D. N. & Itti, L. (2012). Modeling the influence of action on spatial attention in visual interactive environments, IEEE international conference on robotics and automation (ICRA), pp. 444-450.

[35] Achanta, R., Shaji A, Smith, K., Lucchi, A., Fua, P. & Süsstrunk, S. (2012). SLIC superpixels compared to state-of-the-art superpixel methods, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 11, pp. 2274-2281.

[36] Frintrop, S. (2005). VOCUS: A visual attention system for object detection and goal directed search, Ph.D. dissertation, Rheinische Friedrich-Wilhelms-University at Bonn, Bonn, Germany.

[37] Gonzalez, R. C. & Woods, R. E. (2007). Digital Image Processing, 3rd edition, Prentice-Hall, pp. 455-456.

[38] Foolad, S. & Maleki, A. (2016). Salient Regions Detection using Background Superpixels, 24th Iranian Conference on Electrical Engineering (ICEE).

[39] Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N. & Tang, X., and Shum, H. (2011). Learning to detect a salient object, IEEE PAMI.

[40] Borji, A. Sihite, D. & Itti, L. (2012). Quantitative analysis of human-model agreement in visual saliency modeling: A comparative study. IEEE TIP.

نشریه هوش مصنوعی و داده کاوی

مدل برجستگی بصری مبتنی بر گراف با استفاده از رنگ پس‌زمینه

شیما فولاد و علی مالکی*

دانشکده مهندسی برق و کامپیوتر، دانشگاه سمنان، سمنان، ایران.

ارسال ۲۰۱۵/۰۷/۰۳؛ بازنگری ۲۰۱۶/۰۳/۱۲؛ پذیرش ۲۰۱۷/۰۳/۱۵

چکیده:

برجستگی بصری یک مفهوم روانشناسی شناختی است که برخی محرک‌های صحنه را نسبت به همسایه‌هایش برجسته‌تر می‌کند و توجه فرد را به خود جلب می‌کند. محاسبه برجستگی بصری، موضوعی است که اخیرا موردتوجه قرارگرفته است. در اینجا، ما یک روش مبتنی بر گراف برای تشخیص برجستگی ارائه می‌کنیم که حاوی سه مرحله می‌باشد: پیش پردازش، تشخیص برجستگی اولیه و تشخیص برجستگی نهایی. نقشه برجستگی اولیه براساس اختلاف رنگ بین نواحی پس‌زمینه و نواحی دیگر به دست می‌آید. در تشخیص برجستگی نهایی، یک گراف ایجاد می‌گردد و روش رتبه‌بندی به کار گرفته می‌شود. در روش پیشنهادی، پس‌زمینه به طور موثر حذف می‌گردد و اغلب، نواحی برجسته به درستی انتخاب می‌شوند. نتایج تجربی بر روی پایگاه داده MSRA-1000، عملکرد عالی و پیچیدگی محاسباتی کم روش پیشنهادی را در مقایسه با روش‌های دیگران نشان می‌دهد.

کلمات کلیدی: توجه بصری، مدل پایین به بالا، تشخیص برجستگی، مبتنی بر گراف، رنگ پس‌زمینه.