Multimed Tools Appl (2017) 76:25091–25107
DOI 10.1007/s11042-017-4725-7

Salient object detection via multiple saliency weights

Weimin Tan[1] · Bo Yan[1]

Received: 15 September 2016 / Revised: 16 February 2017 / Accepted: 17 April 2017 /
Published online: 18 May 2017

Abstract Salient object detection aims to emulate the extraordinary capability of human visual system, which has the ability to find the most visually attractive objects in a complex visual scene. The human visual attention is often complicated and affected by many factors. In this paper, we present a novel bottom-up approach to automatically detect salient objects of an image via multiple visual cues. The key idea of our approach is to represent a saliency map of an image as an integration of multiple visual cues (saliency weights), which have been proven to be effective and useful. Specifically, we propose four saliency weights, i.e., local contrast weight, superpixel clarity weight, background probability weight, and central bias weight, to effectively represent each visual cue. To obtain our saliency map, the four resulting saliency weights are integrated in a principled way via multiplication and summation based fusion. Furthermore, we propose a new superpixel-level saliency smoothing approach to optimize the integrated results for producing clean and consistent saliency maps. Our experimental results on three standard benchmark datasets demonstrate that the proposed approach outperforms other saliency detection approaches in terms of the subjective observations and objective evaluations.

Keywords Salient object detection · Visual cues · Multiple saliency weights · Saliency smoothing

✉ Bo Yan
byan@fudan.edu.cn

Weimin Tan
wmtan14@fudan.edu.cn

[1] School of Computer Science, Shanghai Key Laboratory of Intelligent Information Processing, Fudan University, Shanghai 201203, China

1 Introduction

Vision is one of the most intensively studied directions in the research of human brain, especially with pre-attentive visual selection behavior, which makes a lot of scientists fascinated, including cognitive neuroscience, neuropsychology, and computer science and so on [22]. Recently, theoretical studies of vision indicate that neural activities in retina and primary visual cortex (V1) represent the saliencies of visual inputs in a bottom-up pattern, such that visual information can be efficiently encoded and selected for further detailed or attentive processing [22, 41]. Saliency of vision, a bottom-up visual process, is highly related to visual uniqueness, differences, clarity, surprise, and so on. Inspired by biological vision, there is a lot of work to exploit image properties such as color, illumination, gradient, edges, the spatial relationship of foreground and background to estimate saliency [2, 8, 10, 19, 33].

Different from general segmentation algorithms partitioning an image into multiple regions with coherent properties, saliency detection aims to identify salient object regions from an image. Since the saliency map can represent the important information in the source image, saliency detection plays a vital role in image understanding, analysis, and processing. It has been applied to a variety of applications including image segmentation [37], object recognition and understanding [31], content-aware image/video retargeting [23] [38], content-based image retrieval [14], and image/video compression [18], etc.

In order to obtain high quality and accurate saliency detection results, a lot of bottom-up works have been proposed to explore the distinction between objects and backgrounds in recent years. We review the representative and relative works that employ various visual cues for salient object detection. For a more comprehensive survey of state-of-the-art in visual attention modeling, we refer readers to [35] and [8].

A widely used saliency cue is the globally statistical features of an image, including color contrast [10], luminance, edges and gradients [19], and spectral analysis [2]. Itti et al. [19] propose to use color contrast for salient region detection. Seo and Milanfar [33] propose a saliency measure called self-resemblance to calculate pixel's saliency. The self-resemblance measure is computed by comparing a pixel to its surroundings. Achanta et al. [2]. propose to compute the saliency of each pixel based on the difference between pixel's color and the average image color. Hou and Zhang [17] propose a fast saliency detection approach based on the discovery of spectral residual. Their approach is able to detect foreground objects in an image without any prior knowledge. Rahtu et al. [30] employ a conditional random field (CRF) model to segment initial saliency map, which is produced by using local feature contrast in illumination, color, and motion information. Since sliding window is used in their approach, the authors exploit integral histogram method and graph cut solvers to improve the computational efficiency.

Besides, the image background information is also exploited in several approaches [21, 42] and has been proven to be a useful saliency clue. Zhu et al. [42] proposed a background measure called boundary connectivity, which is the ratio of the connected boundary length of a superpixel to the superpixel size. Because image segmentation itself is a unsolved problem, it is hard to estimate the size of a superpixel and its connected boundary length. So Zhu et al. [42] proposed a "soft" approach to compute each superpixel size and its connected boundary length. They constructed an undirected graph with edges weighted by the color contrast between neighboring superpixels. The contribution of a superpixel to another one are computed based on the accumulated edge weights along their shortest path on the graph. The "soft" computing approach can solve the estimation problem of superpixel size and its connected boundary length to some extent. However, it fails when the colors of salient object are similar to boundary. Besides, it is not an efficient approach for computing.

The approaches mentioned above usually operate on raw image or video. Recently, several approaches exploit the compressed domain information such as transform coefficients and motion vectors to directly detect the saliency of an image [12] or a video [13] in the compressed domain. Fang et al. [12] is a good example. Fang et al. [12] propose a saliency detection approach in the compressed domain. They calculate four feature maps for a JPEG image based on the image features including intensity, color, and texture. These features are extracted based on the DCT coefficients. The final saliency map for the JPEG image is obtained by integrating these four feature maps. Their approach yields impressive results.

Detecting saliency in crowded scenes is a novel work. Jiang et al. [20] propose an interesting approach for detecting saliency in crowded scenes. Based on the observation that face features play an important role in determining saliency, especially in the context of crowd, the authors extract low-level center-surround contrast and high-level semantic face features for saliency prediction in crowd. To automatically combine these features for predicting saliency in crowd, they use multiple kernel learning (MKL) to learn a classifier from their built eye-tracking dataset [20]. Based on the Random Forest algorithm, Ma et al. [26] propose a new crowd saliency prediction approach by optimizing feature combination. Instead of only using traditional features such as low-level features (i.e., color, intensity, orientation) and face features (i.e., face size, face density, frontal face, profile face), they define two new features, FaceSizeDiff and FacePoseDiff, to improve the quality of saliency detection [26].

Since visual attention is often affected by various factors, we need to consider multiple visual cues simultaneously in order to obtain accurate and robust saliency detection results. In fact, by integrating multiple cues to get the final results, this idea can be seen in a lot of works. Yang et al. [39] is a good example for image quality prediction. Yang et al. [39] propose two novel sub-models to separately process user-generated images, which is a multi-dimensional data including text, image, and social relations. The results of these two models are fused together to generate the final score of quality prediction. The results of their approach indicate that their predicted score is fairly consistent with the ground truth.

In this paper, we propose a novel bottom-up approach to automatically detect salient object regions in an image. Our approach is performed in superpixel level for reducing computations. We first segment the input image into a set of superpixels using superpixel segmentation algorithm [24]. Since the superpixels are often the result of over-segmentation, the regions in the input image having coherent image attributes may be partitioned into multiple independent superpixels. In order to make the representation of regions more compact while further reducing the number of superpexels, we propose to fuse neighboring superpixels with consistent image features such as color and texture. After the input image has been separated into several distinct regions, our goal is to find the salient region in the separated regions. Based on the widely accepted biological visual saliency cues, we propose four saliency weights, i.e., local contrast weight, superpixel clarity weight, background probability weight, and central bias weight, to effectively measure the saliency of each region. The results of these four weights will be integrated together to produce our final saliency map. Furthermore, in order to obtain a clean saliency map that its saliency areas are more consistent with the object regions, we propose a superpixel-level saliency smoothing algorithm to optimize the integrated saliency map. The overview of our approach is presented in Fig. 1. The key contributions of our paper are summarized as follows:

- We propose a superpixel fusion algorithm, which is helpful to reduce the number of superpexels and makes the saliency map more consistent with object. The main idea is to fuse neighboring superpixels with consistent features such as color and texture.

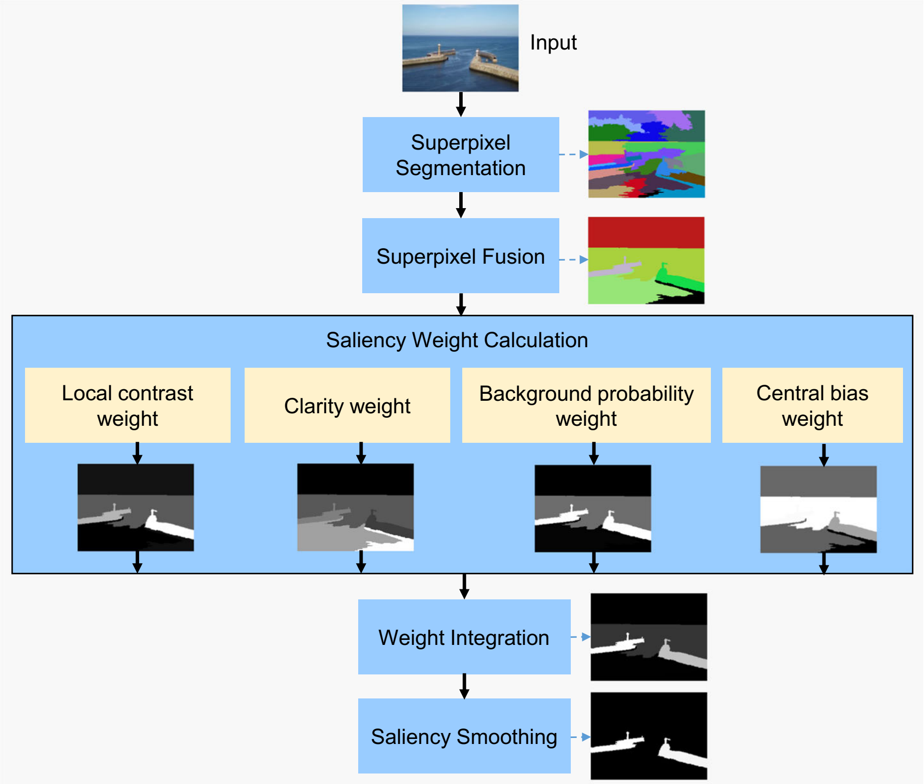


Fig. 1 Procedure of the proposed approach

- We propose four powerful saliency weights that consider clarity of superpixels, spatial informations and color contrast between superpixels. These saliency weights have low computational complexity and are capable of effectively representing visual saliency cues. In addition, the four resulting saliency weights are integrated in a principled way via multiplication and summation based fusion.
- In order to optimize the integrated saliency map obtained above, we propose a superpixel-level saliency smoothing algorithm to make the saliency areas more consistent with the object regions.

In the following sections, we will detail these four saliency weights and the saliency smoothing algorithm, and show how each weight is to determine the saliency of each region. The remainder of this paper is organized as follows. In Section 2, we introduce our saliency detection approach in detail. Section 3 presents the experimental results. An application of our approach is introduced in Section 4. We will draw our conclusions in Section 5.

2 The proposed approach

As Fig. 1 shows, firstly, the input images are segmented into a set of superpixels using a fast and robust superpixel segmentation algorithm [24]. Secondly, we fuse those neighboring superpixels with consistent color and texture features in order to reduce the number of superpixels. Then, based on the theory of human visual attention and the observation of spatial layout difference of image background and foreground, we propose four saliency weights, i.e., local contrast weight, superpixel clarity weight, background probability weight, and central bias weight, to effectively measure the saliency of each fused superpixel. Finally, the final saliency maps are obtained by integrating all saliency weights. Furthermore, to obtain a clean saliency map, we perform saliency smoothing step to optimize the integrated saliency map. Figure 2 illustrates the pipeline of our saliency detection approach. In the following subsections we will describe our approach in detail.

2.1 Superpixel segmentation and fusion

Our approach is performed in superpixel level, so we first segment the input image into a set of superpixels using Liu's algorithm [24] (we use a MATLAB implementation from http://mingyuliu.net/), which is fast and robust for images with different natural scenes. Figure 2b shows the over-segmentation results.

Then, we fuse neighboring superpixels with coherent features. The motivation for performing superpixel fusion is to reduce the impact of superpixel segmentation results on the saliency detection. Besides, fusing the superpixels with coherent features is not only beneficial to improve the computational efficiency, but also makes the final saliency value more

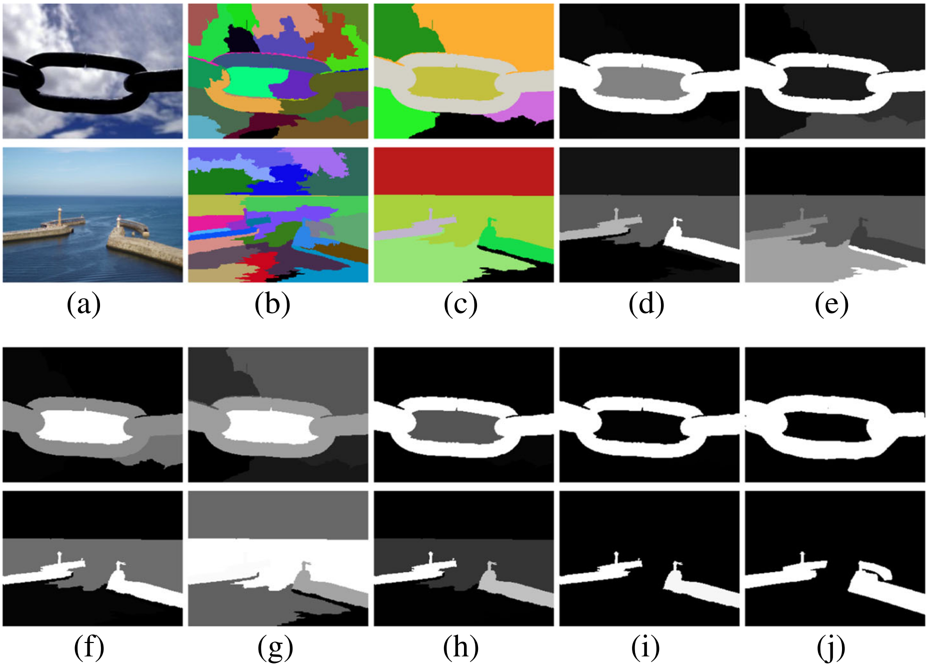


Fig. 2 The pipeline of our approach. (**a**) Input Image, (**b**) Superpixel Segmentation, (**c**) Superpixel Fusion, (**d**) Local Contrast Weight, (**e**) Superpixel Clarity Weight, (**f**) Background Probability Weight, (**g**) Central Bias Weight, (**h**) Integration Weight, (**i**) Superpixel-Level Saliency Smoothing, and (**j**) Ground Truth. It demonstrates that after combining the weights of local contrast, superpixel clarity, background probability, and central bias, we get high quality saliency maps (h and i) comparable to human labeled ground truth

consistent. We use a four-dimensional feature vector to represent each superpixel. Following [15], the feature vector consists of CIE-Lab color and Gabor filter. The Gabor filter responses with 8 orientations. Both the bandwidth and the extracted scale are set to one. The amplitude response of Gabor filter is calculated by combining 8 orientations as the texture feature. When the feature contrast of two neighboring superpixels is less than the threshold T, the two superpixels are fused into a new large superpixels. Therefore, the number of superpixel clusters is determined by the content of the detected image. In the experiment, we observe that the number of superpixel clusters after fusion is about 4∼18. The threshold T is defined as:

$$T = mean(SP_{contr}) - std(SP_{contr})/2 \tag{1}$$

where SP_{contr} denotes the contrast of neighboring superpixels in the CIE-Lab color and texture feature space for the detected image. $mean(SP_{contr})$ and $std(SP_{contr})$ denote the mean and the standard deviation of SP_{contr}, respectively.

Figure 2c shows the results of superpixel fusion. From Fig. 2c we observe that the background consists of only a few superpixels, and the foreground is essentially represented by a new large superpixel. In our approach, we use vector $f = \{f_i\}, i = 1, 2, \ldots, M$, to denote the fused superpixels. M represents the total number of fused superpixels. f_i denotes the i^{th} fused superpixel.

2.2 Saliency weight calculation

Local contrast weight The human visual system pays close attention to the local part of an image. Theories of physiology of vision and neuroscience has proven that the central 10 °C of visual field is represented by at least 60% of the visual cortex and has the greatest visual acuity and color sensitivity [6, 11, 32]. Figure 3 illustrates an example. It shows only a small portion of the image will be processed by the human visual system carefully, while the rests are almost ignored. This conclusion is also accepted by human visual attention theory [3, 19, 21, 34, 36].

Based on the theories of visual field and human visual attention [19, 21, 36], we propose a local contrast approach to calculate the saliency value of each superpixel. The superpixels having more contrast than its surroundings attract more visual attention. This particular superpixel will be selected as the perceptually salient region.

It should be noted that our local contrast approach is different from the widely used global contrast method. Figure 4 shows an illustrative example of global contrast vs. our

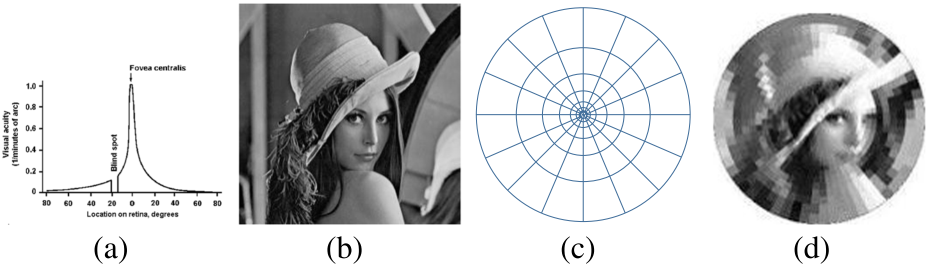


Fig. 3 (**a**) Visual acuity as a function of position on the retina. Note that visual acuity is maximal at 0 °C eccentricity (the central visual field), whereas it is minimal in more peripheral areas [6, 11, 32]. (**b**) Original image. (**c**) Pixel spatial distribution. (**d**) An example of retinal imaging: from the image center to the peripheral areas, the resolution is changed from high to low

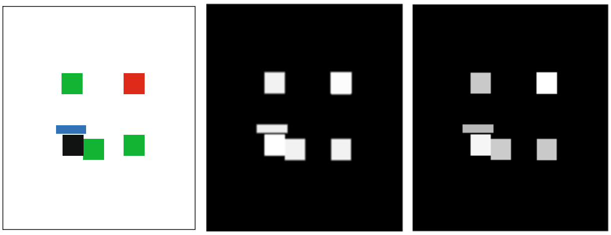

(a) Input Image (b) Global Contrast (c) Our Local Contrast

Fig. 4 Global contrast vs. our local contrast. The *red block* in the input image (*left*) is more salient than the others. If using the global contrast method, the *black block* becomes the most salient object (*middle*), while the detection results of our local contrast is the red block (*right*), which gives more consistent result with the human visual attention

local contrast. In Fig. 4, obviously, the most salient object is the red block instead of the black block in the input image. The possible reason is that the red block is surrounded by the white background in local area, while the black block is surrounded by the white background as well as a green block and a blue block. The saliency contribution of the white area on the top right corner of the black block is greatly reduced because the green block and the blue block are on the top and right side of the black block, respectively.

Specifically, we define the saliency of a superpixel f_i using its feature contrast to its surrounding superpixels in the image. To calculate the local contrast weight, we construct an undirected weighted graph by connecting all neighboring superpixels (f_i, f_j) and assigning their weight $Dist(f_i, f_j)$ as the Euclidean spatial distance between superpixel f_i and f_j. The local contrast weight $W_{LC}(f_i)$ of a superpixel f_i is defined as:

$$W_{LC}(f_i) = \sum_{j=1}^{M} \frac{C_{f_i,f_j} \cdot \sqrt{Size(f_j)}}{Dist(f_i, f_j)} \tag{2}$$

where $Size(f_j)$ is the number of pixels in superpixel f_j. C_{f_i,f_j} is the local feature contrast between superpixel f_i and superpixel f_j. Note that C_{f_i,f_j} is different from feature distance between superpixels f_i and f_j in the CIE-Lab color and texture feature space. C_{f_i,f_j} is computed as:

$$C_{f_i,f_j} = \begin{cases} Contr(f_i, f_j), & if\ (f_i, f_j)\ adjacent \\ Contr(f_i, f_j) - \max\limits_{k \in Path(i,j)} Contr(f_k, f_{k+1}), & if\ (f_i, f_j)\ not\ adjacent \end{cases} \tag{3}$$

where $Contr(f_i, f_j)$ is the feature contrast between superpixels f_i and f_j in the CIE-Lab color and texture feature space. Equation (3) shows that when the superpixel f_i and f_j are not adjacent, the local contrast C_{f_i,f_j} equals $Contr(f_i, f_j)$ minus the maximum feature contrast along their shortest path on the graph. That is to say, the final contrast of the superpixel f_i and f_j should consider not only their feature contrast, but also the feature contrasts in their shortest path on the graph.

Equations (2) and (3) encourage those superpixels with large feature contrast to its surrounding regions. Note that this is quite different from global contrast method which defines the saliency for each region as the weighted sum of the region's contrasts to all other regions in the image [10]. We calculate the shortest paths between all superpixel pairs using algorithm [7]. As our graph is very sparse, computing (3) is efficient and low storage. Figure 2d shows the results of normalized local contrast weight.

Superpixel clarity weight Compared with blur regions, we are usually more interested in objects that are clarity in an image. So the clarity cue should be considered in the calculation of perceptually salient region. The question is how to measure the clarity of a region in an image.

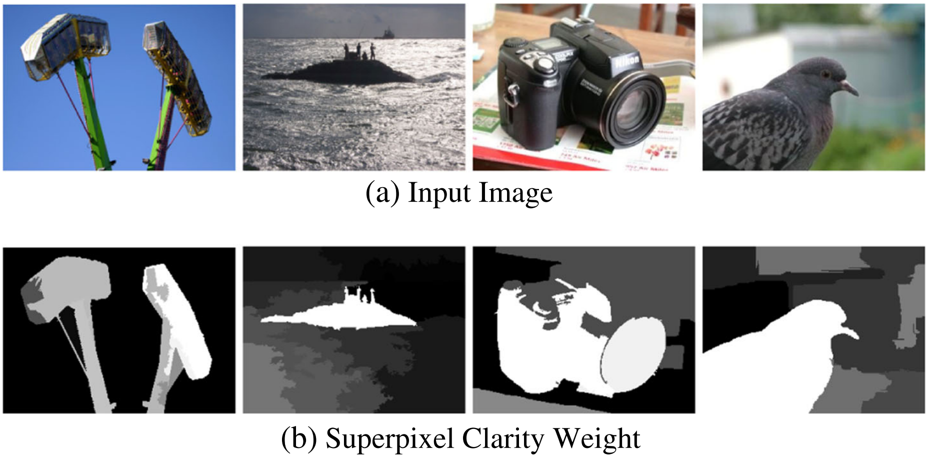
Based on the observation that image clarity is correlated with the image attributes such as richness of edge, contrast, illumination, *etc*. We propose an approach to measure the clarity of each superpixel. The superpixel clarity weight $W_{SC}(f_i)$ of a superpixel f_i is defined as:

$$W_{SC}(f_i) = \frac{Edge(f_i)}{Gray(f_i)} \tag{4}$$

where $Edge(f_i)$ is the average value of color edge of superpixel f_i [16]; $Gray(f_i)$ is the average gray value of superpixel f_i. Equation (4) means that if a region has rich edges and relatively low illumination, its clarity is relatively high. Figure 2e shows the results of normalized superpixel clarity weight. It demonstrates that the importance of each superpixel can be well discriminated according to (4). Figure 5 shows more results of superpixel clarity weight.

Background probability weight Intuitively, background regions are much more connected to image boundaries than foreground ones, i.e., the less the regions touch the image boundary, the more salient they will be. Zhu et al. [42] proposed a "soft" approach to compute the boundary connectivity, which is inefficient and fails when the colors of salient object are similar to boundary. Different from [42], we directly compute each superpixel size and its connected boundary length based on the results of our superpixel fusion. The background probability weight $W_{BG}(f_i)$ of a superpixel f_i is defined as:

$$W_{BG}(f_i) = exp\left(-\frac{\left(\frac{Conbd(f_i)}{\sqrt{Size(f_i)}} - \omega_{bg}\right)^2}{2\sigma_{bg}^2}\right) \tag{5}$$


(a) Input Image

(b) Superpixel Clarity Weight

Fig. 5 More examples of superpixel clarity weight. It clearly demonstrates the importance of each superpixel can be well discriminated according to the superpixel clarity weight

where $Conbd(f_i)$ is the number of pixels in the image boundary of superpixel f_i. The square root of the superpixel size is to achieve image size-invariance. In our implementation, ω_{bg} and σ_{bg} are set to 0 and 0.2, respectively.

Based on the superpixel fusion results, the calculation of (5) is very fast and effective because we only need to count the number of pixels in the image boundary and each superpixel. This is feasible because we experimentally find that the foreground and background in the input image are usually represented by only one or a few superpixels after superpixel fusion operation. Figure 2f shows the results of normalized background probability weight. It shows that the background in the input image can be well depressed according to (5).

Central bias weight In human visual system, the image center regions draw more attention than the other regions [15], i.e., the saliency values of central regions are higher than the image boundary regions. Many works use the central bias as a saliency cue to suppress the background close to image boundary [4, 15, 42]. In this fashion, the central bias weight $W_{CB}(f_i)$ of a superpixel f_i in our notation can be written as:

$$W_{CB}(f_i) = \frac{\sum_{p_k \in f_i} e^{\frac{-Dist(p_k,0)^2}{2\sigma_{cb}^2}}}{Size(f_i)} \tag{6}$$

where $Dist(p_k, 0)$ is the spatial distance between pixel p_k and image center. In our implementation, we set $\sigma_{cb} = 0.5$. Equation (6) shows that the central bias weight of each superpixel is obtained by averaging the central bias weights of pixels in each superpixel. Figure 2g shows the results of normalized central bias weight. It demonstrates that the central bias saliency cue can depress the boundary backgrounds in the input image.

2.3 Weight integration

So far, we have introduced four bottom-up saliency weights. If used independently, each weight has its merits and, of course, demerits. The common integration approaches are linear summation and pixel-wise multiplication of all the saliency weights [15]. Figure 6 shows the difference between summation and multiplication combinations. Generally, multiplication encourages the common saliency regions in each weight and gives the saliency of higher precision. Summation favors to obtain higher recall.

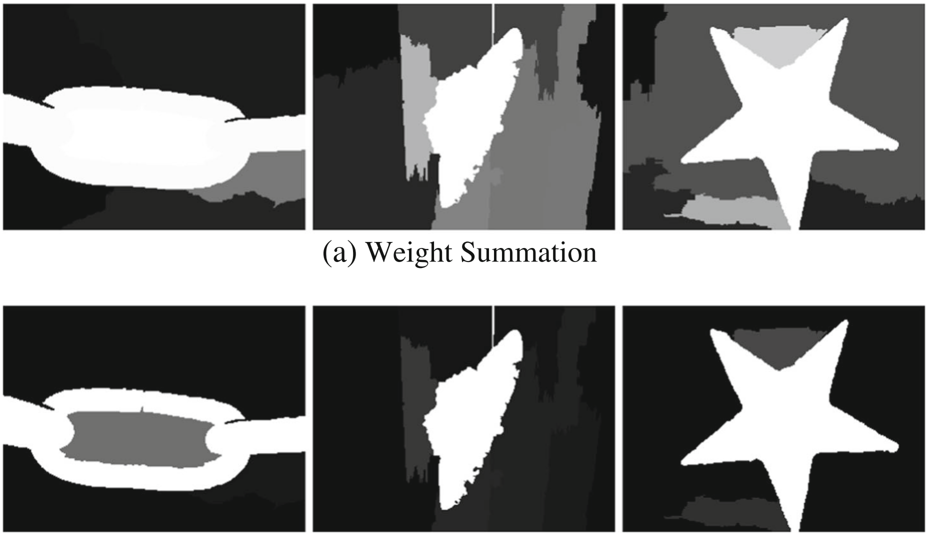
In this paper, we integrate the advantages of multiplication and summation and use the following principle to combine our four saliency weights mentioned above. For a superpixel f_i, the integration of these weights is defined as:

$$S(f_i) = \omega \cdot S_{multi}(f_i) + \varphi \cdot S_{sum}(f_i) \tag{7}$$

$$S_{multi}(f_i) = W_{LC}(f_i) \cdot W_{SC}(f_i) \cdot W_{BG}(f_i) \cdot W_{CB}(f_i) \tag{8}$$

$$S_{sum}(f_i) = \alpha \cdot W_{LC}(f_i) + \beta \cdot W_{SC}(f_i) + \gamma \cdot W_{BG}(f_i) + \lambda \cdot W_{CB}(f_i) \tag{9}$$

where $W_{LC}, W_{SC}, W_{BG}, and\ W_{CB}$ are our four saliency weights. $S(f_i)$ is the integrated weight of the superpixel f_i. In our implementation, ω and φ are set to 0.5, which means that the results of multiplication and summation make the equivalent contribution to the integration result. The parameters in (9) such as $\alpha, \beta, \gamma, and\ \lambda$ are empirically set to 0.5, 0.5, 1, 0.5, respectively. Note that there are some approaches that aim at automatically fusing multiple saliency weights/cues via learning algorithms. In our experiment, instead of using the complex automatic fusion method, we empirically set the integrating parameters in (7) and (9).


(a) Weight Summation

(b) Weight Multiplication

Fig. 6 Weights combination: summation and multiplication. The background noises can be effectively depressed by multiplying each weight, which can improve the saliency detection accuracy. And summation favors to obtain higher recall

2.4 Saliency smoothing

By integrating each saliency weight, we have obtained the saliency map. Based on the observation that the neighboring superpixels with coherent color and texture features should have consistent saliency values, we propose to refine the integrated saliency map by performing superpixel-level saliency smoothing. Specifically, the saliency value of a superpixel is equal to the weighted average of the saliency values of other superpixels. When calculating the smoothed saliency value of a superpixel, we not only consider the feature contrast between the superpixel and other superpixels, but also consider the spatial distance between them.

For a superpixel f_i, the smoothed saliency value $S'(f_i)$ is defined as:

$$S'(f_i) = h\left(\frac{\sum_{j=1}^{M}\left[1 - Contr(f_i, f_j)\right]\left[1 - Dist(f_i, f_j)\right] S(f_j)}{\sum_{j=1}^{M}\left[1 - Contr(f_i, f_j)\right]\left[1 - Dist(f_i, f_j)\right]}\right) \tag{10}$$

$$h(x) = e^{-\frac{(x-\omega_{sm})^2}{2\sigma_{sm}^2}} \tag{11}$$

Equation (10) encourages those superpixels with low feature contrast and small spatial distance to make more contribution for the smoothed saliency value $S'(f_i)$. Equation (11) is used to normalize the smoothed saliency value S', which is computed by (10). In our implementation, ω_{sm} and σ_{sm} are set to 1 and 0.2, respectively.

Figure 2i shows the results of smoothed saliency maps. It illustrates that the non-saliency noises in the weight combination results (Fig. 2h) are obviously reduced. It should be noted that although saliency smoothing operation can optimize the integrated saliency map, the quality of saliency detection depends mainly on the four powerful saliency weights mentioned above.

3 Experimental results

3.1 Experimental setup

Dataset To evaluate our approach, we carried out several experiments on three standard benchmark datasets: MSRA [2], SED1 [4], and SED2 [4]. MSRA [2] consists of 1,000 images with different natural scenes and complex backgrounds. SED1 [4] consists of 100 images with low contrast and cluttered background scenes making it challenging for saliency detection. SED2 [4] contains 100 images with two salient objects. The human-labeled foreground masks used as ground truth for salient object detection in MSRA [2], SED1 [4], and SED2 [4] datasets are also provided.

Evaluation criterion In our experiments, we adopt five criteria to evaluate the quantitative performance of different approaches: receiver operating characteristic (ROC) curve, mean absolute error (MAE) [29], mean precision, mean recall, and F-measure. The ROC curve plots the true positive rate against the false positive rate and presents a robust evaluation of saliency detection performance. Specifically, the ROC curve is obtained by thresholding the saliency map using a series of fixed integers from 0 to 255.

MAE is proposed by [29], which provides a estimate of the dissimilarity between the saliency map and ground truth. It calculates the mean absolute error between the detected saliency map (S) and the binary ground truth (GT). MAE is computed as:

$$MAE = \frac{\sum_{i=1}^{W}\sum_{j=1}^{H}|S(i,j) - GT(i,j)|}{W \times H} \tag{12}$$

We also use F-measure to evaluate the overall performance. F-measure is computed as:

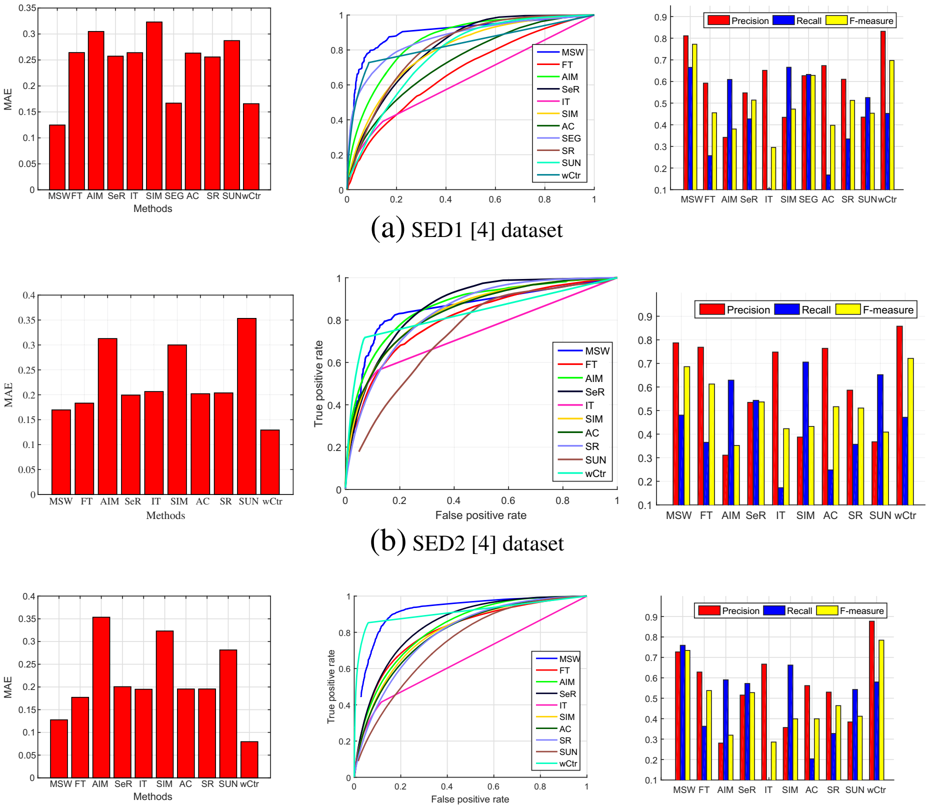
$$F_\gamma = \frac{(1+\gamma^2) \times Precision \times Recall}{\gamma^2 \times Precision + Recall} \tag{13}$$

where precision and recall are an average value which is obtained by averaging a number of precisions of thresholding saliency map. As described in [2, 25], precision is more important than recall for attention detection. Therefore, we use $\gamma^2 = 0.3$ to weigh precision more than recall.

3.2 Comparison of saliency detection approaches

For convenience, we use (MSW) (Multiple Saliency Weights) to represent our multiple-weight integration approach. We compare MSW with the following representative saliency detection methods, including SR [17], IT [19], SIM [27], SUN [40], AC [1], SeR [33], AIM [9], FT [2], SEG [30], and wCtr [42] respectively. These approaches are very typical in saliency detection and implemented using their either publicly available source code or original saliency detection results from the authors.

Figure 7 reports the experimental results of all approaches on the SED1, SED2, and MSRA datasets. The results demonstrate the overall better quality of saliency maps generated by using our MSW approach in terms of the measures of MAE, ROC, and F-measure. Specifically, Fig. 7 (left column) shows the MAE comparison results of all approaches, which indicate that our MSW approach obtains the lowest MAE scores in SED1 [4], SED2 [4], and MSRA [2] datasets, except for wCtr [42] approach on SED2 and MSRA [2].



(a) SED1 [4] dataset

(b) SED2 [4] dataset

(c) MSRA [2] dataset

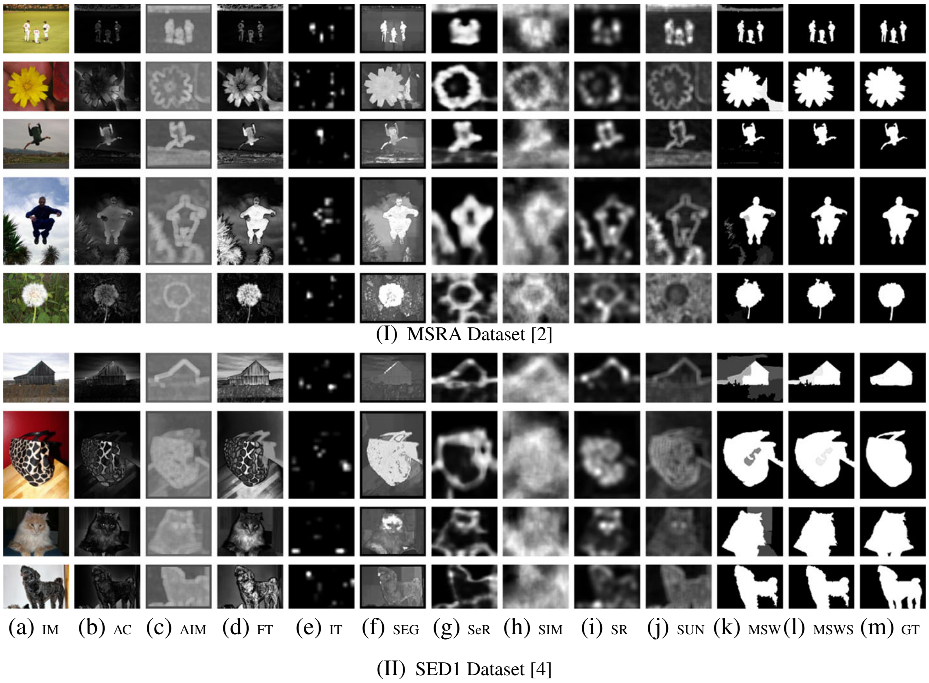
Fig. 7 MAE, ROC curve, and F-measure performance of all the ten approaches on the SED1 [4], SED2 [4], and MSRA [2] datasets. From *top to bottom*: SED1 [4], SED2 [4], and MSRA [2] datasets are tested. From *left to right*: MAE, ROC curves, and F-measure performance are displayed. The experimental results demonstrate the overall better quality of saliency maps generated by using our approach

The experimental results demonstrate that the saliency maps produced by the proposed approach are more consistent with the ground truth.

Figure 7 (middle column) shows the comparison results of ROC curves of our MSW approach and other methods. Since our approach takes into account the multiple visual cues instead of a single cue, our MSW approach reasonably outperforms other competing methods in SED1 [4], SED2 [4], and MSRA [2] datasets. Given a fixed false positive rate, MSW obtains a higher true positive rate than other saliency detecting approaches in most cases.

Furthermore, Fig. 7 (right column) also shows the average F-measure performances of our MSW approach and other methods on the SED1 [4], SED2 [4], and MSRA [2] datasets. The experimental results demonstrate that our MSW approach outperforms other methods in terms of precision, recall, and F-measure on both two standard benchmark datasets in most cases.

Figure 8 presents some visual examples of salient object detection on the MSRA [2] and SED1 [4] datasets for a subjective comparison. It intuitively demonstrates that the saliency


(I) MSRA Dataset [2]

(II) SED1 Dataset [4]

Fig. 8 Visual comparison of saliency detection on the MSRA [2] and SED1 [4] datasets. (**a**) input images (IM), (**b**) - (**j**): saliency maps generated using different approaches, (**k**) our MSW and (**l**) MSWS (Multiple Saliency Weight Smoothing) approaches, and (**m**) ground truths (GT)

maps obtained by our approach provide visually more pleasuring saliency detection results than other competing approaches, and surprisingly, are more close to the ground truth.

In Table 1, we compare the average processing time on SED1 [4] with other saliency detection approaches mentioned above. The processing environment has an Intel® Xeon® CPU with 2.53 GHz operational frequency and 24G bytes RAM size under Windows® Server 2008 operating system. All the algorithms are implemented by MATLAB. Table 1 demonstrates that the time complexity of our MSW approach is relatively low compared with other methods.

4 Application

The result of saliency detection can be used to improve the existing image processing applications. Content-aware image retargeting is a good example. It judiciously retargets

Table 1 Comparison of processing time (seconds per image)

Method	FT [2]	SEG [30]	SR [17]	AC [1]	SIM [27]	MSW
Time(s)	0.21	16.42	0.13	15.18	2.14	0.56

an image to the target resolution based on an importance map for the image [5, 28]. We experiment with using different saliency maps in the image retargeting approach: As Similar-As-Possible (ASAP) [28]. ASAP [28], a typical continuous approach, was proposed by Panozzo et al. [28]. It optimizes a mapping (warping) from the resolution of source image to some target resolution using several types of constraints in order to protect the important contents in the source image.

Figure 9 demonstrates the retargeting results using different saliency maps. The saliency maps are from image gradient, FT [2], IT [19], and our MSW. It intuitively demonstrates that

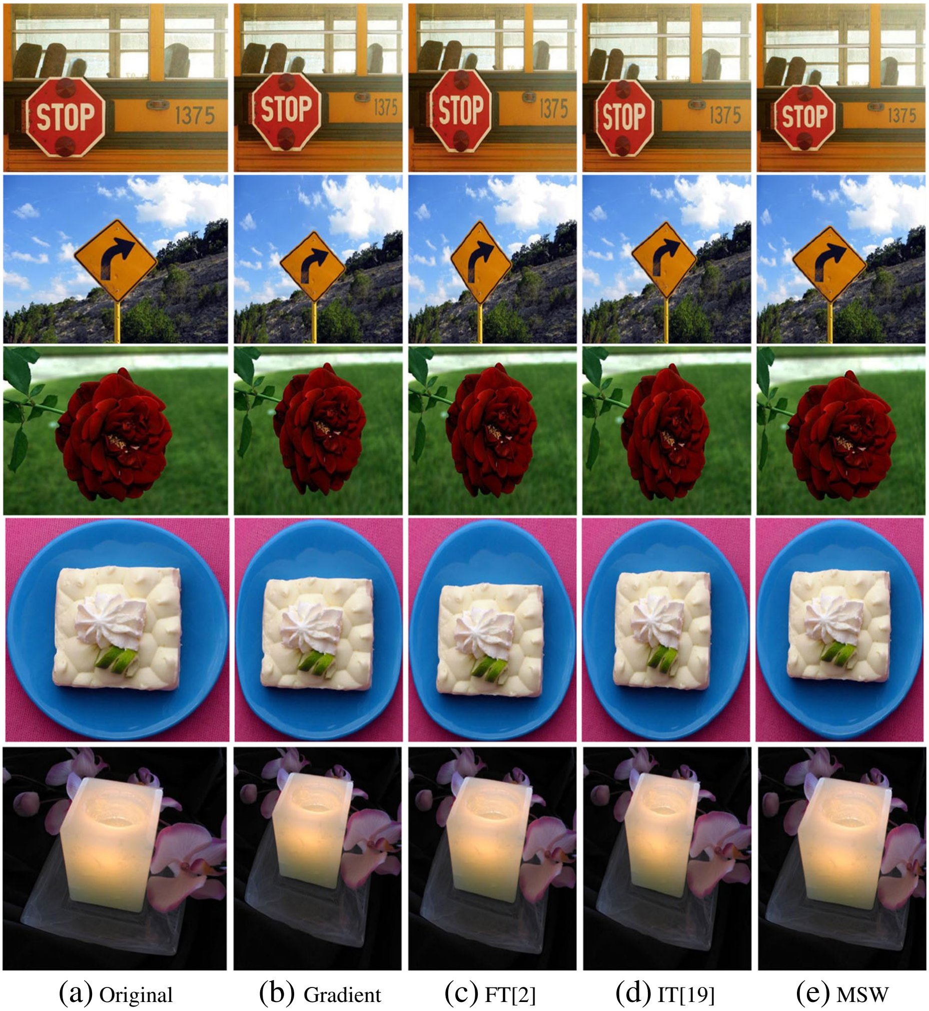


Fig. 9 Image retargeting results (75% original width). Comparison of content-aware image retargeting results [28] using the saliency maps of gradient, FT [2], IT [19], and our MSW

the retargeting approach of ASAP [28] employing our saliency maps is capable of producing better retargeting results. This is reasonable because the saliency maps produced by our approach are consistent with the object regions, which is important for importance map based retargeting approaches. However, gradient maps often have higher saliency values at object boundaries. The saliency maps of FT [2] and IT [19] cover less object regions, which are not suitable for image retargeting.

5 Conclusion

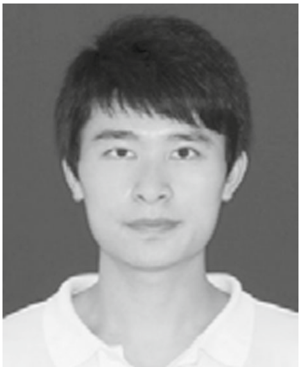
In this paper, we have presented a novel salient object detection approach that estimates the saliency of regions by using our four powerful saliency weights, i.e., local contrast weight, superpixel clarity weight, background probability weight, and central bias weight. The final saliency maps are obtained by integrating these saliency weights. Furthermore, we propose a superpixel-level saliency smoothing approach to optimize the integrated results for obtaining clean and consistent saliency maps. Extensive experiments on three standard benchmark datasets show that our approach achieves good performance and is computationally efficient. In the future, we aim to exploit the extrinsic information (such as the images having visually similar content with the original image) to further improve the performance of our algorithm.

Acknowledgments This work was supported in part by NSFC (Grant No.: 61370158; 61522202).

References

1. Achanta R, Estrada F, Wils P, Ssstrunk S (2008) Salient region detection and segmentation. In: Computer vision systems, vol 5008, pp 66–75
2. Achanta R, Hemami S, Estrada F, Susstrunk S (2009) Frequency-tuned salient region detection. In: IEEE Conference on computer vision and pattern recognition (CVPR'2009), pp 1597–1604
3. Alfano PL, Michel GF (1990) Restricting the field of view: perceptual and performance effects. Percept Motor Skills 70(1):35–45
4. Alpert S, Galun M, Brandt A, Basri R (2012) Image segmentation by probabilistic bottom-up aggregation and cue integration. IEEE Trans Pattern Anal Mach Intell (TPAMI'2012) 34:315–327
5. Avidan S, Shamir A (2007) Seam carving for content-aware image resizing. ACM Trans Graph 26(3):10
6. Barghout-Stein L (1999) On differences between peripheral and foveal pattern masking. Ph.D. thesis, University of California Berkeley
7. Bellman R (1956) On a routing problem. Tech. rep. DTIC Document
8. Borji A, Itti L (2013) State-of-the-art in visual attention modeling. IEEE Trans Pattern Anal Mach Intell 35(1):185–207
9. Bruce N, Tsotsos J (2007) An information theoretic model of saliency and visual search. In: Attention in cognitive systems. Theories and systems from an interdisciplinary viewpoint, vol 4840. Springer, pp 171–183
10. Cheng M, Mitra N, Huang X, Torr P, Hu S (2015) Global contrast based salient region detection. IEEE Trans Pattern Anal Mach Intell (TPAMI'2015) 37:569–582
11. Cline D, Hofstetter HW, Griffin JR (1997) Dictionary of visual science. Butterworth-Heinemann
12. Fang Y, Chen Z, Lin W, Lin CW (2012) Saliency detection in the compressed domain for adaptive image retargeting. IEEE Trans Image Process (TIP'2012) 21(9):3888–3901
13. Fang Y, Lin W, Chen Z, Tsai CM, Lin CW (2014) A video saliency detection model in compressed domain. IEEE Trans Circ Syst Video Technol (TCSVT'2014) 24(1):27–38

14. Fu H, Chi Z, Feng D (2006) Attention-driven image interpretation with application to image retrieval. Pattern Recog 39:1604–1621
15. Fu H, Cao X, Tu Z (2013) Cluster-based co-saliency detection. IEEE Trans Image Process (TIP) 22(10):3766–3778
16. Henriques J (2010) http://www.mathworks.com/matlabcentral/fileexchange/coloredges.m
17. Hou X, Zhang L (2007) Saliency detection: a spectral residual approach. In: IEEE Conference on computer vision and pattern recognition (CVPR'2007), pp 1–8
18. Itti L (2004) Automatic foveation for video compression using a neurobiological model of visual attention. IEEE Trans Image Process (TIP'2004) 13:1304–1318
19. Itti L, Koch C, Niebur E (1998) A model of saliency-based visual attention for rapid scene analysis. IEEE Trans Pattern Anal Mach Intell (TPAMI'1998) 20:1254–1259
20. Jiang M, Xu J, Zhao Q (2014) Saliency in crowd. In: European conference on computer vision (ECCV'2014), pp 17–32
21. Koch C, Ullman S (1987) Shifts in selective visual attention: towards the underlying neural circuitry. In: Matters of intelligence. Springer, pp 115–141
22. Li Z (2014) Understanding vision: theory, models, and data. Oxford University Press
23. Li Z, Qin S, Itti L (2011) Visual attention guided bit allocation in video compression. Image Vis Comput 29:1–14
24. Liu MY, Tuzel O, Ramalingam S, Chellappa R (2011) Entropy rate superpixel segmentation. In: IEEE Conference on computer vision and pattern recognition (CVPR'2011), pp 2097–2104
25. Liu T, Yuan Z, Sun J, Wang J, Zheng N, Tang X, Shum HY (2011) Learning to detect a salient object. IEEE Trans Pattern Anal Mach Intell (TPAMI'2010) 33:353–367
26. Ma K, Gao G, Ding G, Liu CH, Liu E (2016) Crowd saliency prediction with optimal feature combinations. In: Wireless communications & signal processing (WCSP'2016), pp 1–5
27. Murray N, Vanrell M, Otazu X, Parraga C (2011) Saliency estimation using a non-parametric low-level vision model. In: 2011 IEEE Conference on computer vision and pattern recognition (CVPR), pp 433–440
28. Panozzo D, Weber O, Sorkine O (2012) Robust image retargeting via axis-aligned deformation. Comput Graph Forum 31(2pt1):229C236
29. Perazzi F, Krahenbuhl P, Pritch Y, Hornung A (2012) Saliency filters: contrast based filtering for salient region detection. In: IEEE Conference on computer vision and pattern recognition (CVPR'2012), pp 733–740
30. Rahtu E, Kannala J, Salo M, Heikkilä J. (2010) Segmenting salient objects from images and videos. In: Computer vision-ECCV 2010. Springer, pp 366–379
31. Rutishauser U, Walther D, Koch C, Perona P (2004) Is bottom-up attention useful for object recognition? In: Proceedings of the 2004 IEEE computer society conference on computer vision and pattern recognition (CVPR'2004), vol 2, pp II–37–II–44
32. Schmidt RF (1981) Fundamentals of sensory physiology. Springer Science & Business Media
33. Seo HJ, Milanfar P (2009) Static and space-time visual saliency detection by self-resemblance. J Vis 9:15
34. Strasburger H, Rentschler I, Jüttner M. (2011) Peripheral vision and pattern recognition: a review. J Vis 11(5):13
35. Toet A (2011) Computational versus psychophysical bottom-up image saliency: a comparative evaluation study. IEEE Trans Pattern Anal Mach Intell (TPAMI) 33(11):2131–2146
36. Treisman AM, Gelade G (1980) A feature-integration theory of attention. Cogn Psychol 12(1):97–136
37. Wang L, Xue J, Zheng N, Hua G (2011) Automatic salient object extraction with contextual cue. In: IEEE International conference on computer vision (ICCV'2011), pp 105–112
38. Yan B, Li K, Yang X, Hu T (2015) Seam searching-based pixel fusion for image retargeting. IEEE Trans Circ Syst Vid Technol (TCSVT'2015) 25:15–23
39. Yang Y, Wang X, Guan T, Shen J, Yu L (2014) A multi-dimensional image quality prediction model for user-generated images in social networks. Inf Sci 281:601–610
40. Zhang L, Tong MH, Marks TK, Shan H, Cottrell GW (2008) Sun: a bayesian framework for saliency using natural statistics. J Vis 8(7):32
41. Zhaoping L, Zhaoping L (2007) Theoretical understanding of the early visual processes by data compression and data selection. Netw Comput Neural Syst 17(4):301–34
42. Zhu W, Liang S, Wei Y, Sun J (2014) Saliency optimization from robust background detection. In: 2014 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 2814–2821



Weimin Tan received his master's degree at the College of Communication Engineering, Chongqing University, Chongqing, China. He is currently pursuing the doctoral degree with the School of Computer Science at Fudan University, Shanghai, China. His research interests include digital image and video processing.

Bo Yan (Senior Member, IEEE) received his Ph.D. degree in Computer Science and Engineering from the Chinese University of Hong Kong (CUHK) in 2004. Before that, he received his degrees of B.E. and M.E. in Communication Engineering both from Xi'an Jiaotong University (XJTU) in 1998 and 2001 respectively. From 2004 to 2006, he worked in the National Institute of Standards and Technology US (NIST) as a Postdoctoral Guest Researcher. Dr. Yan is currently a Professor in School of Computer Science at Fudan University, Shanghai, China. His research interests include video processing, computer vision and multimedia communications. He has served as the Associate Editor for IEEE Transactions on Circuits and Systems for Video Technology (TCSVT), and the Guest Editor of Special Issue on "Content-aware Visual Systems: Analysis, Streaming and Retargeting" for IEEE Journal on Emerging and Selected topics in Circuits and Systems (JETCAS).