Automatic Salient Object Segmentation Based on Context and Shape Prior

Huaizu Jiang[1]
jianghuaizu@gmail.com
Jingdong Wang[2]
jdwang@microsoft.com
Zejian Yuan[1]
yzejian@gmail.com
Tie Liu[3]
liultie@cn.ibm.com
Nanning Zheng[1]
nnzheng@mail.xjtu.edu.cn
Shipeng Li[2]
spli@microsoft.com

[1] Xi'an Jiaotong University
Xi'an, P. R. China
[2] Microsoft Research Asia
Beijing, P. R. China
[3] IBM China Research Lab
Beijing, P. R. China

Abstract

We propose a novel automatic salient object segmentation algorithm which integrates both bottom-up salient stimuli and object-level shape prior, i.e., a salient object has a well-defined closed boundary. Our approach is formalized as iterative energy minimization framework, leading to binary segmentation of the salient object. Such energy minimization is initialized with saliency map which is computed through context analysis based on multi-scale superpixels. Object-level shape prior is then extracted combining saliency with object boundary information. Both saliency map and shape prior will be updated after each iteration. Experimental results on two public benchmark datasets show that our proposed approach outperforms state-of-the-art methods.

1 Introduction

Human beings own the ability to accurately and rapidly find out the interested object (region), which is called *focus of attention* or *saliency* in a scene. When attention deployment is driven by salient stimuli, it is considered to be rapid, bottom-up, and memory-free. Attention can also be guided by relatively slow, top-down, memory-dependent mechanisms [13]. For instance, when we look at people's faces, those which you are familiar with may draw your attention. Applications for salient object detection include picture collage [27], image retargeting [3], image and video compression [29], and object recognition [23].

Recently many computational models have been proposed for saliency detection. Itti *et al.* [13] computed saliency value for each pixel, i.e., saliency map, based on color and orientation information using "center-surround" operations akin to visual receptive fields. Liu *et al.* [18] proposed several saliency features and integrated them into CRF framework to

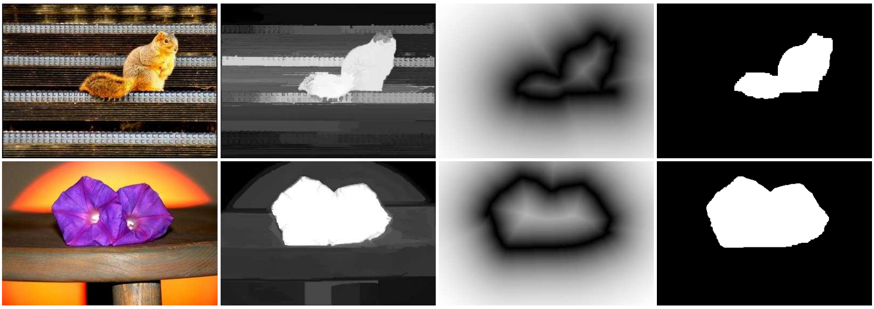


Figure 1: Overview of our proposed salient object segmentation algorithm. Given input images (left), local context analysis can output full-resolution saliency map (middle left, section 2.2). We then extract shape prior (middle right, section 2.3) based on saliency map. Iterative energy minimization based on saliency map and shape prior is employed, leading to accurate binary segmentation of input images (right, section 3).

separate salient object from background. Hou and Zhang [12] and Achanta *et al.* [1] defined saliency in the frequency domain. Goferman *et al.* [11] proposed one kind of saliency which incorporates context information into final saliency map. Recent work of Cheng *et al.* [7] computed saliency map based on regions to ease computational burden.

Of all these works, the most related to ours is [7], since we also compute saliency based on regions for efficiency. Our proposed saliency feature, however, defines a region's saliency w.r.t. its context, i.e., neighbors, instead of all regions in the image, and we incorporate location prior into saliency computation. We also extend our single-superpixel-scale to multiple scales to make our algorithm more robust under complicated circumstances. Finally, we propagate saliency from regions to pixels to get saliency map. See examples in Fig. 2, where our method can uniformly highlight the salient object even in cluttered background.

Our work differs from the previous works mentioned above mostly because we incorporated the generic knowledge of object into salient object segmentation. Recent years, several kinds of object-level prior have been studied. Vicente *et al.* [26] proposed connectivity prior, which assumes that parts of the object are connected together. And Veksler [25] presented the star prior with the assumption that the center of an object is known. Alexe *et al.* [2] proposed a generic objectness measure by combining several image cues to quantify the possibility for an image window to contain an object of any categories. Inspired by [2], we impose the object-level prior, that the object has a well-defined closed boundary, on our salient object segmentation algorithm. But unlike [2], which defines the closure on a rectangle, we straightly search for such a closed contour. Our computed salient contour combines saliency with boundary information, defined as a ratio form suggested by Stahl and Song in [24] which can be efficiently optimized by using the ratio contour algorithm proposed by Song *et al.* [28]. Object-level shape prior can then be extracted based on such optimal contour. See sample images of shape prior in Fig.1.

Computed optimal contour is actually a polygon (see section 2.3 for details). Since we prefer pixel-wise segmentation, we choose to integrate shape prior into energy minimization framework as a constraint instead of directly outputting it like [30]. Energy minimization has been greatly adopted in image segmentation [5, 16, 22]. Initial saliency and shape prior are only rough estimation of the salient object, thus we re-estimate both of them after each iteration and re-segment the image. Unlike [22], which only updates appearance model in iterative energy minimization, both of our appearance and shape model evolve in our framework.

We have evaluated our proposed algorithm on two publicly available datasets provided by Achanta *et al*. [1] and Liu *et al*. [18], respectively. And we compared our approach with other state-of-the-art methods [1, 7, 11, 12, 13]. Experimental results show that we can achieve better performance on both datasets.

This paper is organized as follows. Section 2 introduces context-based saliency and object-level shape prior computation based on three characteristics of a salient object. Section 3 describes the iterative energy minimization framework. We show experimental results in Section 4 to demonstrate the effectiveness of our proposed approach. Sections 5 concludes the paper.

2 Salient Object Features

In this section, we first introduce three characteristics of the salient object. According to these characteristics, we compute saliency map and object-level shape prior. Compared with previous works [1, 7, 12, 13, 18], which only take bottom-up salient stimuli into consideration, our approach incorporates object-level shape information to better define a salient object.

2.1 Three Characteristics of a Salient Object

Based on observation, we introduce three characteristics to define a salient object:

1. The salient object is always different from its intermediate context.
2. The salient object in an image is most probably placed near the center of the image.
3. A salient object has a well-defined closed boundary.

The first characteristic, based on bottom-up salient stimuli, has been extensively studied in previous works [11, 12, 13, 18]. The second one as a location prior, has been studied in photo quality assessment [4, 8, 20], known as *Rule of Thirds*. The rule indicates that to attract people's attention, the object of interest, or main element in a photograph should lie at one of the four intersections to approximate the "golden ratio" (about 0.618). And the last one is satisfied by all categories of objects, as a generic knowledge of an object proposed in [2]. Such a constraint will be incorporated into the energy minimization framework (Section 3) to improve the performance of our proposed salient object segmentation.

2.2 Context-based Saliency Computation

In this section, we introduce context-based saliency computation according to characteristics 1 and 2 of the salient object.

Our saliency is defined based on the superpixels, which are generated by fragmenting the image [9]. One benefit to define saliency upon region is related to efficiency [7]. The previous works [11, 18] resize the original image to a smaller size in order to ease the heavy computational burden. Since the number of superpixels in an image is far smaller than the number of pixels, computing saliency at region level can significantly reduce the computation. We thus can produce full-resolution saliency map.

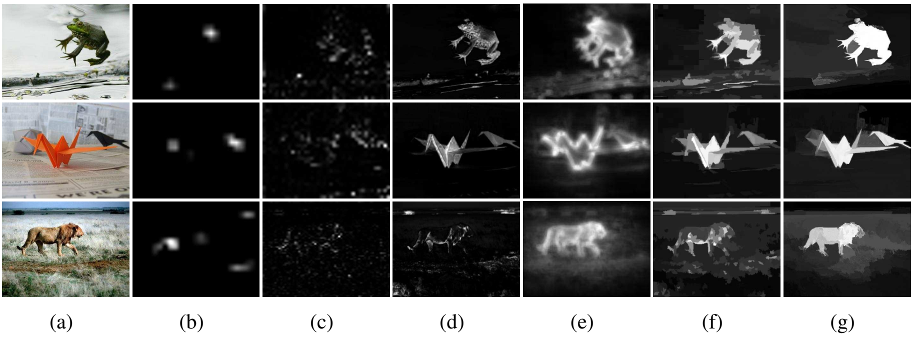


Figure 2: Visual comparison of saliency maps. (a) input image, (b) saliency maps using the method of Itti [13], (c) Hou and Zhang [12], (d) Achanta *et al.* [1], (e) Goferman *et al.* [11], (f) Cheng *et al.* [7], and (g) our method. Our method generates full-resolution saliency maps, in which the salient objects are uniformly highlighted.

Without knowledge of the size of the salient object, previous works [2, 11, 13, 17, 18] employed Gaussian pyramid, a common multi-scale approach for saliency detection. Following the intrinsic idea but we adopt a different method to detect a salient object on multiple superpixel scales, which is obtained by fragmenting the image with N groups of different parameters.

According to characteristic 1, a region (superpixel) is salient if it is distinguished from its immediate context, defined as a set of spatial neighbors in our scenario. Specifically, at superpixel scale n we first fragment input image I into regions $\{r_i^{(n)}\}_{i=1}^{R(n)}$. Given region $r_i^{(n)}$, and its spatial neighbors $\{r_k^{(n)}\}_{k=1}^{K(n)}$, the saliency of $r_i^{(n)}$ is defined as:

$$S(r_i^{(n)}) = -w_i^{(n)} \log \left(1 - \sum_{k=1}^{K(n)} \alpha_{ik}^{(n)} d_{color}(r_i^{(n)}, r_k^{(n)}) \right), \tag{1}$$

where $\alpha_{ik}^{(n)}$ is the ratio between the area of $r_k^{(n)}$ and total area of the neighbors of $r_i^{(n)}$. $d_{color}(r_i^{(n)}, r_k^{(n)})$ is the color distance between regions $r_i^{(n)}$ and $r_k^{(n)}$, computed as χ^2 distance between the CIE L*a*b* and hue histograms of two regions. According to characteristic 2, we introduce the Gaussian falloff weight, defined as $w_i^{(n)} = \exp(-9(dx_i^{(n)})^2/w^2 - 9(dy_i^{(n)})^2/h^2)$, where w, h are the width and height of the image respectively, and $(dx_i^{(n)}, dy_i^{(n)})$ are the average spatial distance of all pixels in $r_i^{(n)}$ to image center.

Finally, we propagate saliency value from multiple regions to pixels. Saliency of pixel p is defined as:

$$S_m(p) = \frac{\sum_{n=1}^{N} \sum_{i=1}^{R(n)} S(r_i^{(n)}) \left(||I_p - c_i^{(n)}|| + \varepsilon \right)^{-1} \delta \left(p \in r_i^{(n)} \right)}{\sum_{n=1}^{N} \sum_{i=1}^{R(n)} \left(||I_p - c_i^{(n)}|| + \varepsilon \right)^{-1} \delta \left(p \in r_i^{(n)} \right)}, \tag{2}$$

where i is the index of region, n is the index of superpixel scale, ε is a small constant (0.1 in our implementation), $c_i^{(n)}$ is the color center of region $r_i^{(n)}$, $||I_p - c_i^{(n)}||$ is the color distance from the pixel p to the color center of $r_i^{(n)}$, and $\delta(\cdot)$ is the indicator function.

Another reason why we detect the salient object in multiple scales is related to the scale of context. In [18], the center-surround color histogram feature is computed by comparing the

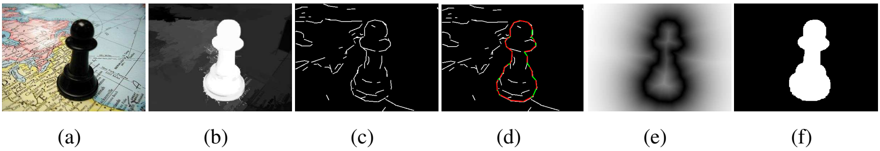


Figure 3: An illustration of shape prior extraction. (a) input image, (b) saliency map, (c) detected line segments using *Pb* edge detector [21], (d) detected optimal contour, where red line means detected segments and green lines are gap-filling segments, and (e) extracted shape prior according to Eqn. 4, (f) binary segmentation result.

candidate salient region's appearance with its context, where the scale of context is assumed to be the same as the object. And in [2], the scale of context for all images is fixed and learned from training data. Here we propose detecting the salient object with respect to multiple scales of the context. Extension to multiple scales will make our saliency computation more robust in a complicated environment and thus achieve better performance. Code of our saliency map computation is available at https://sites.google.com/site/jianghz88/.

2.3 Shape Prior Extraction

In this section, we show how to extract shape prior, i.e., a salient closed contour, which combines saliency with boundary information. Our goal is to extract a closed contour C, which covers the salient object. Specifically, as shown in Fig. 3, we first construct an edge map E. The edge map consists of a set of line segments as illustrated in Fig. 3(c), which are obtained from an edge detector, followed by a line fitting step. We refer to straight line segments as *detected segments*. Note that a detected segment may come from the boundary of the salient object, or the noise and texture of the object and background.

Our shape prior extraction can then be formalized to find an optimal closed contour C^* by identifying a subset of detected segments in E and connecting them together. Since the detected segments are disjoint, we construct additional line segments that fill the gaps between detected segments to form closed contours. We refer to these as *gap-filling segments*. Without knowing which gaps are along the resulting optimal contour, we construct a gap-filling segment between each possible pair of the endpoints of the different detected segments. In this way, a closed contour is defined as a cycle that traverses a set of detected and gap-filling segments alternately, as shown in Fig. 3(d). The optimal closed contour C^* can be defined as:

$$C^* = \arg\min_{C} \frac{|C_G|}{\sum_{p \in C} S_m(p)}, \tag{3}$$

where $|C_G|$ is the total length of gaps along the contour C, and $\sum_{p \in C} S_m(p)$ is the total saliency value of pixels located inside C. Ratio contour algorithm [28] can be employed to find such an optimal cycle in polynomial time.

Finally, the shape prior S_p is defined as

$$S_p(p) = 1 - \exp\left(1 - \gamma d(p)\right), \tag{4}$$

where $d(p)$ is the spatial distance between pixel p and optimal closed contour C^*, computed by using a distance transform [10], as shown in Fig. 3(e). And γ is the confidence of the shape prior, set to 1 in our implementation. Note that computed optimal contour is actually a polygon. Since we prefer pixel-wise segmentation, unlike [30] which straightly outputs

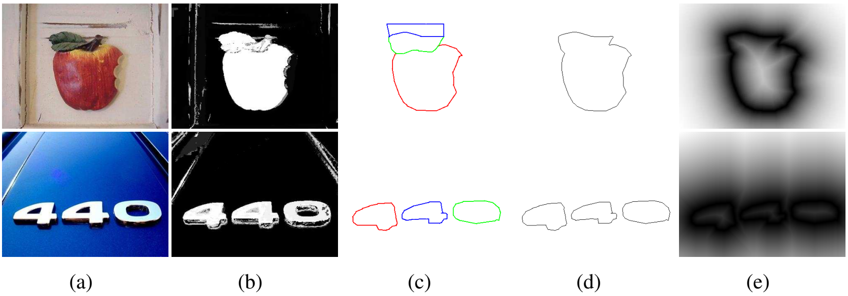


Figure 4: More examples of shape prior extraction. (a) input image, (b) re-estimated saliency map (see section 3.2 for details), (c) all detected contours with different color (red, green, blue in 3 iterations respectively), (d) merged optimal contour (see text for details), and (e) extracted shape prior.

the optimal contour, we choose to integrate shape prior into the energy minimization framework for binary salient object segmentation (section 3). See sample images in Fig. 3, the segmented mask has more accurate and smooth boundary than the optimal contour.

An object may consist of several parts and each can be represented by a closed boundary. For instance, the apple shown in the first row of Fig. 4(a) consists of an apple and its leaf, both of which can be represented by a closed boundary as shown in Fig. 4(c). In addition, there may be several objects in one image, as shown in the second row of Fig. 4(a). Therefore, we search for n_c (set to 3 in our implementation) contours. After getting one optimal contour, we simply set the saliency value inside it to zero instead of removing corresponding segments like [30]. Then we re-run the ratio contour algorithm. Contours which have self-intersection and whose average saliency value is smaller than T_s (set to 0.65 in our implementation) are rejected. And two contours will be merged if they the share same segments or if one is inside another one. For example, in the first row of Fig. 4, both the apple and its leaf can be successfully detected in the first two iterations (shown in red and green color, respectively), and a noisy contour is also extracted in the last iteration (shown in blue color). The last extracted noisy contour is rejected since its average saliency value is lower than T_s. By detecting multiple contours we can more accurately identify the salient object.

3 Salient Object Segmentation Framework

Our salient object segmentation framework combines bottom-up saliency information with object-level shape prior. Based on the initial rough estimation, segmentation can be solved by energy minimization. More accurate saliency map and shape prior can then be re-estimated from the new segmentation.

3.1 Energy Model for Salient Object Segmentation

Given input image I, saliency map S_m (section 2.2), and shape prior S_p (section 2.3), our goal is to find the label set L, where $l_p \in \{0,1\}$ for each pixel p, 0 for background and 1 for salient object (foreground). Salient object segmentation can be formalized as energy minimization:

$$E(L) = \sum_{p \in \mathscr{P}} U(p, l_p, S_m) + \lambda \sum_{(p,q) \in \mathscr{N}} \delta(l_p \neq l_q) V(p, q, I, S_p), \tag{5}$$

where $\mathscr{P}$ is the set of image pixels and $\mathscr{N}$ is a 4-connected neighbor system. The data term U is defined as

$$U(p,l_p,S_m) = \begin{cases} S_m(p), & l_p = 0 \\ 1 - S_m(p), & l_p = 1 \end{cases}. \tag{6}$$

The smoothness term can be written as

$$V(p,q,I,S_p) = \alpha V_a(p,q,I) + (1-\alpha) V_s(p,q,S_p), \tag{7}$$

where α (set to 0.5 in our implementation) controls the relative importance of two parts. $V_a(p,q,I)$ is defined as the traditional smoothness term [18, 22]: $V_a(p,q,I) = \exp\left(-\frac{||I_p - I_q||^2}{2\beta}\right)$, where $\beta = \mathbf{E}\left(||I_p - I_q||^2\right)$ as in [22]. $V_s(p,q,S_p)$ can be derived from the shape prior, defined as:

$$V_s(p,q,S_p) = S_p(\frac{p+q}{2}) \approx \frac{S_p(p) + S_p(q)}{2}. \tag{8}$$

Intuitively, this term encourages the segmentation boundary to be aligned with computed closed contour. According to [14], such energy can be efficiently minimized by using the min-cut/max-flow algorithms proposed in [6], leading to a binary segmentation of the image.

3.2 Iterative Energy Minimization

The initial saliency map and shape prior are only rough estimation of the salient object. After binary segmentation, both of them can be re-estimated more accurately. Unlike previous work [7], our iterative energy minimization framework updates both appearance and shape models.

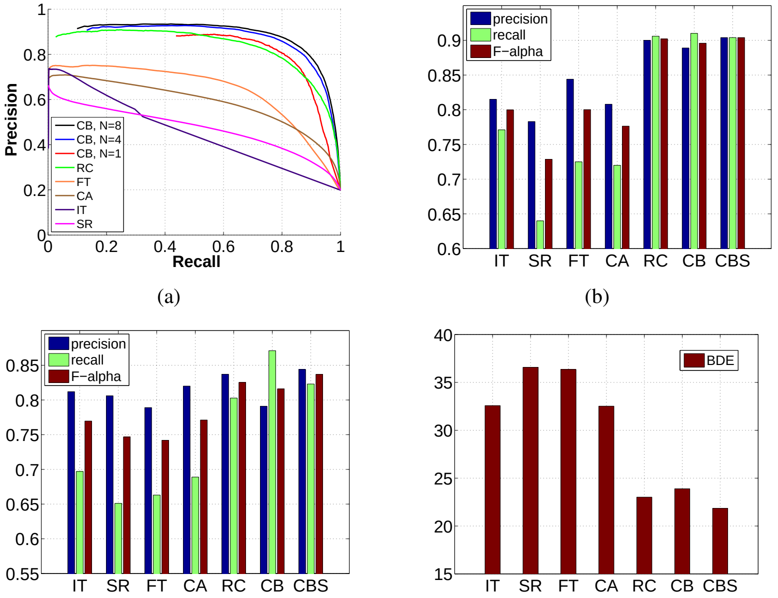
Specifically, we construct CIE L*a*b* and HSV histograms H_F, H_B for salient object (foreground) and background, based on current segmentation, respectively. To make reliable estimation, we dilate the mask of current segmentation to get a trimap. Region outside dilated region is set as background, and inside region will be set as salient object (foreground). Then the updated saliency map can be defined as

$$S_m(p) = \frac{H_F(b_p)}{H_F(b_p) + H_B(b_p)}, \tag{9}$$

where b_p is the color histogram bin of pixel p. The less overlap between appearance of foreground and background, the more accurate the updated saliency map is. Based on such a new saliency map, we can update the shape prior, then re-segment the image. We run iterative energy minimization until convergence (at the most 4 iterations in our implementation). The algorithm of our iterative segmentation is summarized below:

Algorithm 1 L=SalientObjectSegmentation(I)

```
1: Calculate saliency map $S_m$ according to Eqn. 2.
2: Extract shape prior $S_p$ based on $S_m$ according to Eqn.4.
3: Segment image through energy minimization according to Eqn. 5.
4: Update saliency map $S_m$ based on current segmentation $L$ according to Eqn. 9.
5: Go to step 2 to update shape prior $S_p$, and then re-segment image until convergence.
```



(a)

(b)

(c)

(d)

Figure 5: Quantitative comparison of different methods on the dataset [1, 18]. See the text for detailed explanation.

4 Experimental Results

We perform experiments on two datasets. The first one is provided by Achanta *et al.* in [1] which contains 1000 images, along with ground truth for each image in the form of accurate human-labeled masks for salient object. And the second one is the MSRA dataset B provided by Liu *et al.* in [18], which contains 5000 images, along with bounding box annotation of salient object for each image. The first dataset is a subset of the second one but with more accurate annotation. Nine user annotations for each image in the second dataset are provided, however, making it more objective for comparison.

To smooth the computed superpixels, we first merge those neighbouring regions whose d_{color} is less than 0.2. To construct the edge map, we use the *Pb* edge detector [21], and the line approximation package provided by Kovesi [15]. We remove all edges with length less than 10 pixels, and set the allowed maximum deviation between an edge and its fitted line segment to 2 pixels.

Our proposed approach is compared with five state-of-the-art saliency detection methods, including IT [13], SR [12], FT [1], CA [11] and RC [7]. IT is a classical approach that leverages a neuromorphic model simulating which elements are likely to attract visual attention. SR and FT work in frequency domain to find the anomalies of an image. CA is a recently proposed method which integrates context information into the final saliency map. And RC is the approach most related to ours, which computes saliency based on a region's global contrast w.r.t. all other regions in an image on a single superpixel scale.

Two experiments are conducted to comprehensively evaluate the performance of our approach to salient object segmentation. In the first experiment, we compare saliency maps produced with different methods since saliency map may be used in many applications, e.g., picture collage [27], image retargeting [3]. In the second experiment, we compare the salient

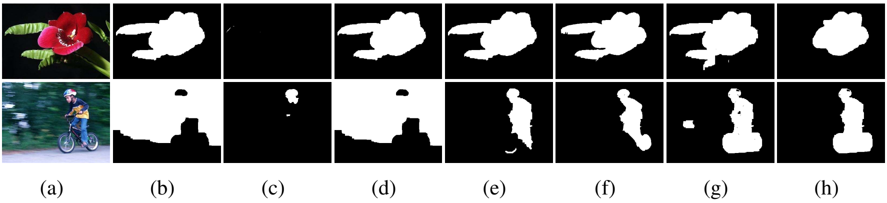


Figure 6: Visual comparison of salient object segmentation using different methods. (b)-(f) are the results of IT [13], SR [12], FT [1], CA [11] and RC [7] respectively. (g) is result of CB, which considers our context based saliency map. And (h) is result of CBS, our proposed approach, which combines context based saliency and object-level shape prior.

object segmentation results of different methods. And we provide comparisons to show the effectiveness of our object-level shape prior.

On the dataset of [1], we computed precision, recall, F_α with $\alpha = 0.5$, to quantitatively evaluate the performance. On the MSRA dataset B, to output a rectangle for evaluation, we exhaustively search for a smallest rectangle containing at least 95% foreground pixels in the binary segmentation as [19]. In addition to precision, recall and F_α, we provide BDE (Bounding box Displacement Error) for bounding box comparison.

4.1 Comparison of Saliency Maps

To compare saliency map, with saliency value in the range [0, 255], we threshold the saliency map at each T_f within [0, 255]. T_f is varied from 0 to 255, and precision and recall are computed at each value of T_f.

We compare our context-based (CB) saliency map with state-of-the-art methods. In addition, to show the effectiveness of our proposed multi-superpixel-scale saliency enhancement, we provide comparisons of our saliency map with different scales (N in Eqn.2). Recall-precision curves are shown in Fig. 5(a). As it shows, we can achieve great improvement from 1-scale to 4-scale enhancement. And the gap between 4-scale and 8-scale is smaller. We therefore choose $N = 8$ in our next experiment. Our 4-scale and 8-scale saliency maps consistently outperform other five state-of-the-art methods. Visual comparison of salient maps is provided in Fig. 2. As we can see, our method can generate better saliency maps. For example, in the last row, our method uniformly highlights the salient object even in cluttered background.

4.2 Comparison of Salient Object Segmentation

In this section we will compare salient object segmentation performance of different methods. As IT, SR, FT and CA evaluate only saliency maps, we use their saliency maps to initialize our iterative segmentation algorithm to make an objective comparison. And for RC we directly report their best result on the same dataset. In addition, we also present the segmentation result by using our saliency map but without shape prior (α is set to 1 in Eqn. 7) to demonstrate the effectiveness of object-level shape prior.

As we can see in Fig. 5(b)(c)(d), our approach integrating context-based saliency and shape prior (CBS) consistently outperforms IT, SR, FT, CA on both datasets. And we can achieve as good result as RC on the first dataset. But we can achieve slightly better performance on the second larger dataset. To achieve binary segmentation, RC first thresholds the saliency map, and then iteratively apply GrabCut [22]. It is difficult to select the initial threshold, however. As we can see, the selected threshold, which gives 95% recall rate in the first saliency map comparison experiment, works quite well on the first dataset. While on a larger dataset, this threshold proves to work poorly. Our method is straightly initialized with saliency map, therefore performs better on the larger dataset.

In addition, we present the segmentation result of CB, which only takes context-based saliency into consideration, to demonstrate the effectiveness of the object-level shape prior. As can be seen, by incorporating shape prior, we can achieve slightly better segmentation precision and F_α on both datasets.

We also provide visual comparisons of salient object segmentation in Fig. 6. In the image of the first row, the segmentation can easily be affected by the leaves, since both flower and leaves are all quite different from background. By incorporating shape prior, however, we can get satisfying segmentation results. And in the second row, the background is cluttered and the salient object consists of several colors. In such a challenging case, IT and FT completely fail, and SR, CA and RC can only find part of the object. Result of CB contains part of the background. And our proposed approach, CBS, successfully segments the salient object.

5 Conclusion

In this paper, we propose context-based saliency and object-level shape prior computation according to the three characteristics of a salient object. Saliency map is computed based on multi-scale superpixels, which proves to significantly enhance saliency, through context analysis. And object-level shape prior is extracted combining saliency with object boundary information. We then integrate both of them into an iterative energy minimization framework, leading to binary segmentation of the salient object, where shape prior encourages segmentation boundary to be aligned with salient contour. The major difference between our approach and previous works is that we take such an object-level prior into consideration to better define a salient object. Experimental results on two benchmark datasets show that our approach can outperform state-of-the-art methods.

Acknowledgements

This work was performed when Huaizu Jiang was an intern at Microsoft Research Asia. And this work was supported by the National Basic Research Program of China under Grant No. 2007CB311005 and the National Natural Science Foundation of China under Grant No. 90820017. The first author would also like to thank Prof. Yanju Liu for grammar checking.

References

[1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. *IEEE CVPR*, 0:1597–1604, 2009.

[2] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. What is an object? In *CVPR*, pages 73–80, 2010.

[3] Shai Avidan and Ariel Shamir. Seam carving for content-aware image resizing. volume 26, 2007.

[4] Subhabrata Bhattacharya, Rahul Sukthankar, and Mubarak Shah. A framework for photo-quality assessment and enhancement based on visual aesthetics. In *ACM Multimedia*, pages 271–280, 2010.

[5] Yuri Boykov and Marie-Pierre Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *ICCV*, pages 105–112, 2001.

[6] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26:1124–1137, September 2004. ISSN 0162-8828.

[7] Ming-Ming Cheng, Guo-Xin Zhang, Niloy J. Mitra, Xiaolei Huang, and Shi-Min Hu. Global contrast based salient region detection. In *IEEE CVPR*, pages 409–416, 2011.

[8] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Ze Wang. Studying aesthetics in photographic images using a computational approach. In *ECCV (3)*, pages 288–301, 2006.

[9] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[10] P.F. Felzenszwalb and D.P. Huttenlocher. Distance transforms of sampled functions. In *Cornell Computing and Information Science TR2004-1963*, 2004.

[11] Stas Goferman, Lihi Zelnik-Manor, and Ayellet Tal. Context-aware saliency detection. In *CVPR*, pages 2376–2383. IEEE, 2010.

[12] Xiaodi Hou and Liqing Zhang. Saliency detection: A spectral residual approach. In *CVPR*, 2007.

[13] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans. P.A.M.I.*, 20(11):1254–1259, 1998.

[14] Vladimir Kolmogorov and Ramin Zabih. What energy functions can be minimized via graph cuts? *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(2):147–159, 2004.

[15] P. D. Kovesi. MATLAB and Octave functions for computer vision and image processing. Centre for Exploration Targeting, School of Earth and Environment, The University of Western Australia. Available from: <http://www.csse.uwa.edu.au/∼pk/research/matlabfns/>.

[16] Yin Li, Jian Sun, Chi-Keung Tang, and Heung-Yeung Shum. Lazy snapping. *ACM Trans. Graph.*, 23(3):303–308, 2004.

[17] Feng Liu and Michael Gleicher. Region enhanced scale-invariant saliency detection. In *ICME*, pages 1477–1480, 2006.

[18] Tie Liu, Jian Sun, Nan-Ning Zheng, Xiaoou Tang, and Heung-Yeung Shum. Learning to detect a salient object. *CVPR*, 0:1–8, 2007.

[19] Tie Liu, Zejian Yuan, Jian Sun, Jingdong Wang, Nanning Zheng, Xiaoou Tang, and Heung-Yeung Shum. Learning to detect a salient object. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(2):353–367, 2011.

[20] Yiwen Luo and Xiaoou Tang. Photo and video quality evaluation: Focusing on the subject. In *ECCV (3)*, pages 386–399, 2008.

[21] David R. Martin, Charless C. Fowlkes, and Jitendra Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *PAMI*, 26:530–549, 2004.

[22] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut – interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23:309–314, 2004.

[23] Ueli Rutishauser, Dirk Walther, Christof Koch, and Pietro Perona. Is bottom-up attention useful for object recognition. In *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR*, pages 37–44, 2004.

[24] Joachim S. Stahl and Song Wang. Edge grouping combining boundary and region information. *IEEE Transactions on Image Processing*, 16(10):2590–2606, 2007.

[25] Olga Veksler. Star shape prior for graph-cut image segmentation. In *Proceedings of the 10th European Conference on Computer Vision: Part III*, ECCV '08, pages 454–467, Berlin, Heidelberg, 2008. Springer-Verlag. ISBN 978-3-540-88689-1.

[26] Sara Vicente, Vladimir Kolmogorov, and Carsten Rother. Graph cut based image segmentation with connectivity priors. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:1–8, 2008.

[27] Jingdong Wang, Jian Sun, Long Quan, Xiaoou Tang, and Heung yeung Shum. Picture collage. In *In Proceedings of CVPR 06*, pages 347–354, 2006.

[28] Song Wang, Toshiro Kubota, Jeffrey Mark Siskind, and Jun Wang. Salient closed boundary extraction with ratio contour. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(4): 546–561, 2005.

[29] J. Xue, C. Li, and N. Zheng. Proto-object based rate control for jpeg2000: An approach to content-based scalability. *IEEE Trans. on Image Processing*, 20(4):1177–1184, April 2011.

[30] Zhiqi Zhang, Yu Cao, Dhaval Salvi, Kenton Oliver, Jarrell Waggoner, and Song Wang. Free-shape subwindow search for object localization. In *CVPR*, 2010.