Salient Object Detection using Concavity Context

Yao Lu, Wei Zhang, Hong Lu and Xiangyang Xue
School of Computer Science, Fudan University, Shanghai, China
{yaolu, weizh, honglu, xyxue}@fudan.edu.cn

Abstract

Convexity (concavity) is a bottom-up cue to assign figure-ground relation in the perceptual organization [18]. It suggests that region on the convex side of a curved boundary tend to be figural. To explore the validity of this cue in the task of salient object detection, we segment the images in a test dataset into superpixels, and then locate the concave arcs and their bounding boxes along boundary of superpixels. Ecological statistics indicate that such bounding box contains salient object with a large probability.

To utilize this spatial context information, i.e. concavity context, we follow the multi-scale analysis of human visual perception and design a hierarchical model. The model yields an affinity graph over candidate superpixels, in which weights between vertices are determined by the summation of concavity context on different scales in the hierarchy. Finally a graph-cut algorithm is performed to separate the salient and background objects. Evaluation on MSRA Salient Object Detection (SOD) dataset shows that concavity context is effective, and our approach provides improvement over state-of-the-art feature-based algorithms.

1. Introduction

Salient object refers to foreground object which attracts more visual attention [15]. Detecting salient object can be formulated as a binary labeling problem aiming to assign figural and background regions. In the perceptual organization, this process is called figure/ground organization (FGO) [18]. It is a high level cognition process which has been studied by the Gestalt psychologists since 1920s. Although many rules for the FGO have been identified, including low-level feature cues like contrast [24] and spatial frequency, geometric cues like symmetry and convexity [13], as well as semantic cues like object familiarity [19], etc.[1], mechanism of FGO is still complex and not certain.

Fortunately, inspired by the neuronal architecture of the early primate visual system, Itti and his colleagues [12] cleverly simplified the process of FGO and merely focus on low-level features to compute bottom-up attention. In their model, center-surround difference of multiple features are incorporated into a dynamic neural network, which makes it possible to determine where in an image the object with visual saliency might occur. Following their opinion, low-level features such as contrast and color become the first choice for many state-of-the-art saliency detection algorithms [31, 14, 8, 32, 2, 15, 16, 9, 11]. For instance, Liu et al. [15] take advantage of a conditional random field to combine features in local, regional and global scales. The features consist of multi-scale contrast, center-surround histogram and color spatial distribution. Achanta et al. [2] successfully perform a rapid frequency-domain analysis using low-level features of color and luminance.

However, detecting low-level feature saliency is not the only way to FGO. In this paper, we utilize a different cue of convexity (concavity), rather than low-level features, to detect salient object. This generic shape cue is proposed by Gerbino et al. [13]. Their experiments on human trials us-

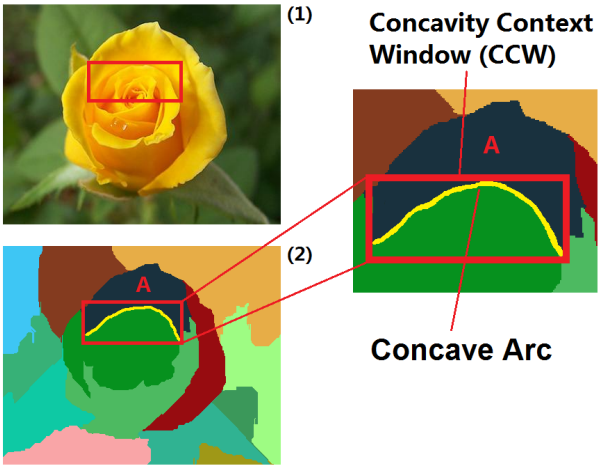


Figure 1. (1) Original image. (2) Segment using [6]. The curve in yellow is one concave arc of superpixel *A*. The rectangle in red is bounding box of the arc (CCW). Concavity context thus encodes the occurrence of salient (figural) and background objects in CCWs.

[1]For more information about FGO, please see http://www.scholarpedia.org/article/Figure-ground_perception

2011 IEEE International Conference on Computer Vision
978-1-4577-1102-2/11/$26.00 ©2011 IEEE

ing some elaborately designed image patterns (stimuli) suggest that, region on the convex side of a curved boundary tend to be figural. Later in the literature, Fowlkes et al. [7] conduct experiments on hand-segmented images by defining convexity as the probability that a line segment connecting two points in a region lies completely within the region. They prove that convexity has the discriminative power of assigning to which region a contour will belong. Ren et al. [22] further design a model to cluster local shapes along object contours. They believe patterns of clustering centers reveal the figure-ground relation. Although the pioneer works have successfully explored the feasibility of generic shape cues in the FGO under some circumstances, there's still a distance from practice. We wonder if this cue is still valid on real-world images that are not perfectly segmented. Furthermore, our purpose is to classify regions and detect salient objects, which is more than to assign contours in [7, 22].

Therefore, we first oversegment the images into superpixels and thus detect *concave arcs* for the contour of each superpixel (see Figure 1 for example). Concave arc is the incurved part of a superpixel boundary with local maximum concavity (convexity on the opposite). Different from [7], we use it to measure the figure-ground relation. After that, we define *Concavity Context Window (CCW)* as bounding box of the concave arc. The *concavity context* thus encodes the occurrence of salient (figural) and background objects inside CCWs. In Section 2, ecological statistics show that about 81% of the CCWs contain salient objects on a test dataset, comparing with the random chance of 60% on average. This reveals that the cue of convexity (concavity) also bias the figure-ground relation on non-perfectly segmented images, and such concavity context information is very helpful to the detection of salient object.

The rest of the paper is organized as follows. In Section 2, a statistical study on the concavity context is introduced in detail. The framework of our approach is described in Section 3. Section 4 evaluates our approach and compares with several feature-based algorithms. Our paper ends in Section 5 by giving a conclusion.

2. Ecological Statistics on Concavity Context

For a given concave arc e, there are three kinds of concavity context inside its corresponding CCW R (see Figure 2 for example):

1. Both sides of e are salient (figural) objects (F-F).
2. Both sides of e are background objects (G-G).
3. There are different objects on the two sides of e (F-G).

We randomly choose 240 images from MSRC-21 [28], PASCAL VOC 2008 [5], and SUN [29] (80 images from each dataset). These three datasets have different complexity of figure-ground relations. We use Felzenswalb's graph-cut algorithm [6] to segment the images into 6-8 superpixels. The concave arc detection algorithm is then performed along the contour of each superpixel and we totally obtain 1,218 concave arcs and corresponding CCWs (see Section 3.2 for the concave arc detection algorithm). We manually classify the detected CCWs into the three concavity context and obtain the statistical result in Table 1.

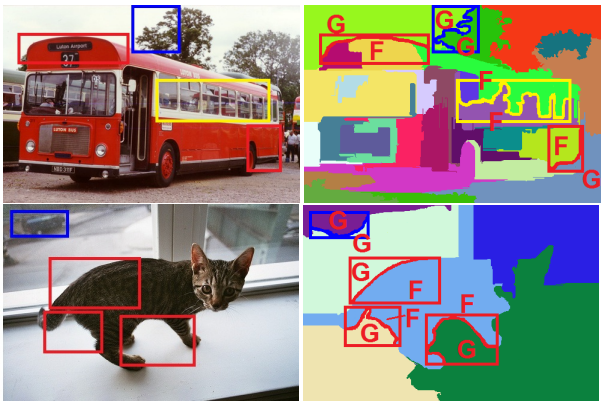


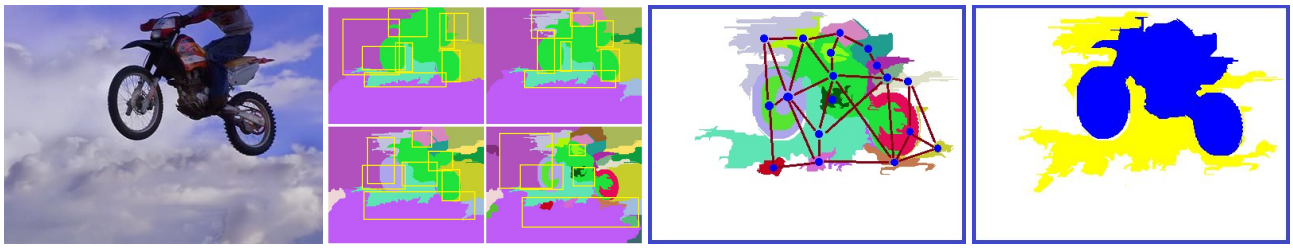
Figure 2. Examples of CCWs. Left: origin images. Right: segment using [6]. F: salient (figural) object, G: background object. Rectangles in (1) yellow: F-F; (2) blue: G-G; (3) red: F-G.

Dataset	MSRC	VOC08	SUN	Avg
Case 1. F-F	48/19%	114/21%	112/26%	22%
Case 2. G-G	46/19%	124/23%	69/16%	19%
Case 3. F-G	154/62%	297/56%	254/58%	59%
Total #	248	535	435	/
Avg CCW Size	29%×27%	21%×22%	23%×21%	/
Chance	66%	58%	57%	60%

Table 1. Row 1-4: Frequency of the 3 concavity context to appear inside detected CCWs. Row 5: Average CCW size (% of image width × % of image height). Row 6: chance to appear figural object within random observation windows. The statistics demonstrate that 81% (Case 1+3) of the CCWs contain salient (figural) object, comparing with the chance of 60% on average.

Furthermore, for each dataset, we calculate the chance to appear salient object inside random observation windows. Such window will count when salient object covers more than 10% of the window area. Intuitively, this chance is in direct proportion to the size of the salient object and the observation window. So for each dataset, we choose the average CCW size as the random observation window size for fair comparison.

The statistical result demonstrates that on average 81% (Case 1+3) of detected CCWs contain salient object, while about 60% of random observation windows contain salient object. Hence, it is clear, convexity (concavity) is also valid under such general condition, and what's more important, *once we find a CCW, for a large probability it contains salient object*.



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

Figure 3. Flowchart of our approach. (a) Input image. (b) Compute hierarchical segmentations and detect concave arcs (CCWs). (c) A weighted graph is built on the candidate superpixels. Notice not all the edges are drawn in this example. (d) Cluster to obtain final result.

3. Approach

3.1. Overview

We build a hierarchical model to separate salient and background objects using the concavity context information. The framework is also capable of detecting multiple salient objects. We briefly describe the framework as follows.

Step 1. Preprocessing. Compute hierarchical segmentations for the input image. Find all concave arcs and CCWs on the multiple segmentations and thus form a candidate superpixel set.

Step 2. Model construction. Build a weighted graph on the candidate superpixels and calculate the distance between vertices.

Step 3. Clustering. Bi-partition the graph and obtain single salient object detection result. Recursively bi-partition the foreground subgraph and obtain multiple/hierarchical salient object detection result.

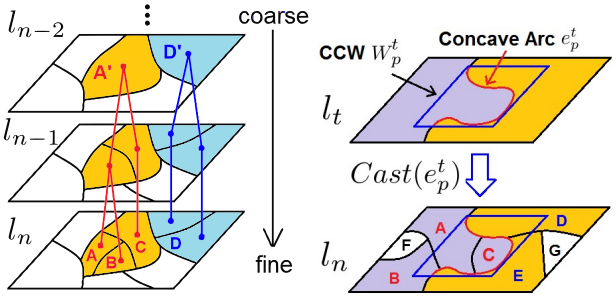
Figure 3 demonstrates the flowchart of our approach.

3.2. Preprocessing

Hierarchical segmentation. Firstly, we compute n hierarchical segmentations by varying the number of superpixels using Felzenswalb's [6] segmentation algorithm.

There are several reasons to employ hierarchical segmentation. First, theoretically it accords with the human visual perception for the multi-scale analysis of objects and scenes [18]. We use this model to add up concavity context on different scales in the hierarchy. Second, experimentally, as one kind of multiple segmentation, it substantially improves spatial support estimation compared to a single segmentation [17, 10, 25]. Finally, superpixels in current layer can be split into several smaller ones in lower (finer) layers, thus we are able to approximately measure the distance between superpixels using the hierarchy (see Figure 4(a)).

Concave arc and CCW detection. Next, we detect all the concave arcs and CCWs in the hierarchical model. The reason for not detecting convex ones is that, by observation we find concave arcs have stronger discriminative power to assign figure-ground relations than convex ones. Moreover, most superpixels are convex at global scale, hence technically finding concave arc is easier.



(a) (b)

Figure 4. (a) We design a hierarchical model based on the multiple segmentations. Two neighboring superpixels will be merged in upper layer if they are similar in color and texture. Suppose $n = 3$, l_{n-2} is the coarsest layer. So original distance between A,B is 1, between A,C is 2, and between A,D is $2+D_{dijkstra}(A', D') = 3$. (b) For the concave arc e_p^t in l_t, $t < n$, we make a cast to the finest layer l_n. The cast concave arc splits the superpixel set $Cast(e_p^t)$ into two parts, and they are shown in different colors.

Previous works in [4, 23] discuss to detect convex and concave parts of digital curves. However their algorithms are sensitive to the selection of starting points, and lack the information of how "concave" the curve is. In this paper, we propose a simple method to find concave parts of digital curves with deep curvature. The main algorithm is demonstrated in Figure 5 and described as follows:

1. For a given superpixel, a B-spline curve algorithm is used to smooth the contour and reduce noise.

2. A minimum enclosing rectangle is drawn for the superpixel and the contour is thus split into four sections.

3. For each section, a main direction is defined (clockwise, 8-connected). Given a pixel p on the curve, we consider $p + 1$ the pixel next to p. The direction of p moving to $p + 1$ is the direction of p. Hence we scan the pixel movements in sequence, and pixel that moves against correct order for more than th steps are regarded as starting point for

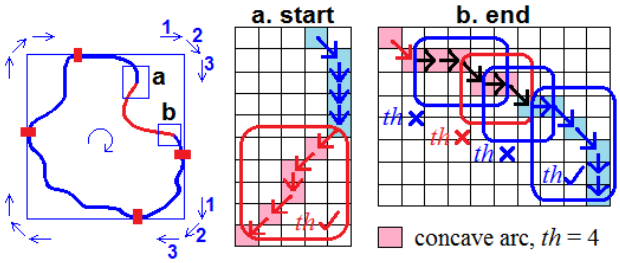


Figure 5. Left: the smoothed superpixel contour is split into four sections and main directions are defined for each section (blue arrows on the corner, in sequence). Mid: pixels which move against main direction, is a sign to start a concave arc (red rectangle). Right: pixels which move in main direction, is a sign to end a concave arc (blue rectangle). Black arrow: pixel movements indicate start or end of a concave arc, but step length does not reach threshold th. The curve in red is one detected concave arc.

a concave arc. We use threshold th here to reduce noise and determine how deep the curvature will be. For example in Figure 5, the section containing the red curve starts from the top and ends in the right. Hence the main direction is right, right-bottom and bottom in sequence. Pixels that move in other directions, or disobey this order for more than 4 steps, are considered as beginning of one concave arc.

4. If starting point of a concave arc is detected, we continue to scan the following pixel movements until we meet correct directions (main direction 1,2,3, or 2,3 in sequence, must also last for th steps). Hence a concave arc is found.

5. Finally concave arcs not long enough will be removed from the result set.

We define $L = \{l_1, l_2, \ldots, l_n\}$ as the set of hierarchical segmentations. l_1 is the coarsest segmentation and l_n the finest. $S^t = \{s_1^t, s_2^t, \ldots, s_{m_t}^t\}$ is the set of superpixels in l_t. $E^t = \{e_1^t, e_2^t, \ldots, e_{k_t}^t\}$ is the set of concave arcs in l_t, and $Win^t = \{W_1^t, W_2^t, \ldots, W_{k_t}^t\}$ is the corresponding CCW set. After concave arc and CCW detection in the hierarchical segmentations, we obtain $E = \{E^1, E^2, \ldots, E^n\}$, and $Win = \{Win^1, Win^2, \ldots, Win^n\}$.

Concave arc and CCW cast. For each concave arc and corresponding CCW detected, we make a pixel-wise cast to the finest segmentation (see Figure 4(b)). This results in the summation of concavity context on different scales in the hierarchical model.

After the cast, concave arcs and CCWs do not change in position, and the CCWs may cover many smaller superpixels. We define these covered superpixels as $Cast(e_p^t)$. At this time, the cast concave arc separates $Cast(e_p^t)$ into two parts. We *do not* care about the order and denote one part as $S_{e_p^t}^+ = \{s_1^n, s_2^n, \ldots, s_u^n\}$, and the other part as $S_{e_p^t}^- = \{s_1^n, s_2^n, \ldots, s_v^n\}$. For example in Figure 4(b), $Cast(e_p^t) = \{A, B, C, D, E\}$, $S_{e_p^t}^+ = \{A, B, C\}$, and $S_{e_p^t}^- = \{D, E\}$.

Selection of candidate superpixels After the above two steps, we regard the union of all the superpixels in $\bigcup_{1 \le t \le n, 1 \le p \le |E^t|} Cast(e_p^t)$ as the candidate superpixel set for model construction.

3.3. Model Construction

Yu and Shi propose a framework to perform figure-ground segregation in [30]. In their model, the Ncut cost function is modified so as to take both bottom-up feature saliency and top-down object familiarity information into consideration. Similarly in our approach, the bottom-up cue of convexity (concavity) is used. But we pay more attention on the construction of the affinity graph, instead of the modification of cost functions.

Let $G = (V, E)$ be an undirected weighted graph, in which V is composed by the candidate superpixels. We define the affinity between superpixel s_i^n and s_j^n as:

$$W(s_i^n, s_j^n) = D(s_i^n, s_j^n) \times (1 + \gamma)^c$$

where $D(s_i^n, s_j^n)$ is the original distance encoding the feature and spatial distance simultaneously between superpixel s_i^n and s_j^n. As discussed in Section 3.2, if two neighboring superpixels appear to be similar in color and texture, they will probably be merged in a coarser layer. We can measure distance between them by finding their least common ancestor (LCA) in the hierarchy. The step length to their LCA is the approximate original distance. Meanwhile, if two superpixels do not have LCA, the original distance then equals to the summation of $n - 1$ and the Dijkstra distance of their cast superpixels on the coarsest layer. n is the total number of layers in the multiple segmentations. Figure 4(a) shows an example to calculate the original distance D.

The term $(1 + \gamma)^c$ encodes the distance adjustment from detected concave arcs. Generally, the aim of distance adjustment is that, superpixels on the same side of the concave arc should be grouped together, and superpixels on different sides of the concave arc should be separated. Thus for each detected concave arc e_p^t, the strategy is:

For each pair in $\{(s_i^n, s_j^n) | s_i^n \in S_{e_p^t}^+, s_j^n \in S_{e_p^t}^-, (s_i^n, s_j^n) \in S_{e_p^t}^+ \times S_{e_p^t}^-\}$, we multiply $W(s_i^n, s_j^n)$ by $(1 + \gamma)$. So that $c => c + 1$;

For each pair in $\{(s_i^n, s_j^n) | s_i^n, s_j^n \in S_{e_p^t}^+, (s_i^n, s_j^n) \in S_{e_p^t}^+ \times S_{e_p^t}^+ \vee s_i^n, s_j^n \in S_{e_p^t}^-, (s_i^n, s_j^n) \in S_{e_p^t}^- \times S_{e_p^t}^-\}$, we divide $W(s_i^n, s_j^n)$ by $(1 + \gamma)$. So that $c => c - 1$.

For example in Figure 4(b), the distance adjustment on e_p^t implies that we make superpixels within $S_{e_p^t}^+ = \{A, B, C\}$ closer to each other, superpixels within $S_{e_p^t}^- = \{D, E\}$ closer to each other, and superpixels between $S_{e_p^t}^+$ and $S_{e_p^t}^-$ further apart from each other.

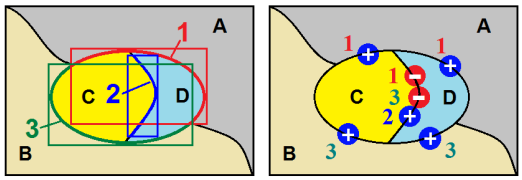


Figure 6. Left: suppose the ellipse (C,D) in the middle is the figural object, and three concave arcs and CCWs are detected. Right: "+" (resp. "-") indicates that superpixels on the two sides of that curve are made further apart (resp. closer) from each other. The numbers aside correspond to the CCWs in the left image. Because of the closure of objects, distance adjustments inside the same object seldom has effect on the distance between different objects. It is clear in this image, the figural object (C,D) can be well grouped together and separated from the background.

Putting everything together, for the calculation of $W(s_i^n, s_j^n)$, we first let $c = 0$ and compute the original distance $D(s_i^n, s_j^n)$. Then we find all the CCWs in the hierarchical segmentations which cover these two superpixels after the cast to l_n. Next we update c using the distance adjustment method stated above for each CCW found. Finally we obtain the c value and $W(s_i^n, s_j^n)$.

The reason for distance adjustment is that, from the statistical result in Table 1, we can see 59% (F-G) of the concave arcs are located at border of salient objects. Hence, (i) over half of all the adjustments (F-G) will be made to enlarge the distance between salient and background objects; (ii) for the rest part of distance adjustments within the same objects (F-F, G-G), they seldom have effect on the distance between salient and background objects because of the closure of objects. This is illustrated in Figure 6. Finally, after summation of all concavity context and distance adjustments on different scales in the hierarchical model, salient and background objects can be largely separated using graph-cut algorithms.

3.4. Clustering

Once the graph model is built, we perform a normalized graph-cut algorithm [26] to regroup the candidate superpixels into salient and background objects.

Single salient object detection. For this task, the clustering of salient and background objects is a binary problem. After bipartition of the graph, we judge which part is salient object by estimating the relation of surroundedness between the two parts, because surroundedness is also a geometric cue indicating that the region surrounded by other regions tend to be figural [18].

Multiple salient object detection. Previous work in [15] discusses the multiple salient object detection. They regard this task as a hierarchical object detection and provide a simple method to detect multiple objects that are disjoint from each other. However there's no such restriction in our framework. We can perform hierarchical/multiple object detection simply by recursively partitioning the foreground subgraph, because the distance adjustment on F-F concavity context separates different objects or parts, which acts in the same way like the figure/ground separation. In Figure 12, we show some good results of our multiple salient object detection method.

Notice that, for a fully unsupervised algorithm, the number of objects/parts should be automatically determined. In the algorithm of normalized cut, the variable $Ncut$ is the cut cost as a fraction of the total edge connections to all the nodes in the graph [26]. We simply use this variable to estimate the number of objects/parts by setting threshold of $Ncut$.

4. Experiments

4.1. Single Salient Object Detection

MSRA SOD dataset (image set A) [15] is a high quality image database for the evaluation of salient object detection algorithms. This dataset contains 5,000 carefully labeled images by multiple users with high labeling consistency. However the ground truth of MSRA SOD is in bounding-box style. Recently in the literature Anchanta et al. [2] further provide the image annotations in object-contour style for a subset of 1,000 images in this dataset. Hence, in our experiments, we evaluate our proposed method using this subset of images.

We follow the scheme of precision, recall and F-Measure to evaluate the accuracy of our method.

4.1.1 Parameter Estimation

Two important parameters in our framework are n and γ. n is the number of layers in the multiple segmentations, and γ is the coefficient in the distance measurement between superpixels, which implies how much the distance will be adjusted for a detected concave arc. Figure 7 shows the F-measure results using different values of n and γ. The experiment is running on 100 randomly selected images. We set $\beta^2 = 1$ to weigh precision and recall the same.

According to the result, it is clear that when $n = 7$ and $\gamma = 0.05$, $F_{\beta^2=1}$ is the highest. Hence in the evaluations below, we choose $n = 7$ and $\gamma = 0.05$.

Moreover, the increase of n significantly improves our detection accuracy, which experimentally proves the study in [17]. Notice when $\gamma = 0$, no distance adjustment is made. Under this circumstance, performance drop down significantly. This means the distance adjustment plays an important role in the clustering procedure.

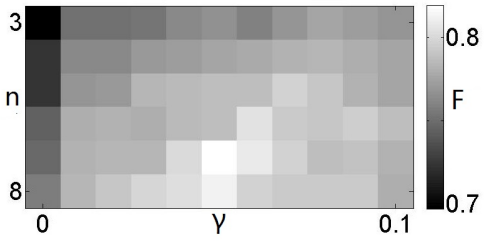


Figure 7. Distribution of $F_{\beta^2=1}$ under different n and γ. When $n = 7, \gamma = 0.05$, $F_{\beta^2=1}$ is the highest.

4.1.2 Quantitative and Qualitative Evaluation

We compare our method with some state-of-the-art feature-based saliency detection algorithms. They are IT [12], MZ [16], GB [9], SR [11], AC [1], IG [2]. We choose $\beta^2 = 0.3$ to follow the experimental settings of the baselines. Figure 8 shows the comparison. Our method achieves comparable performance with the leading method [2], which shows that concavity context is effective, and our hierarchical model is suitable for the task of salient object detection.

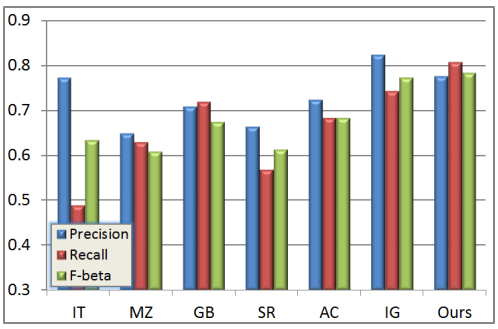


Figure 8. Precision-Recall bars for the evaluation on the dataset and groundtruth in [2]. The baseline data is also reported in [2]. β^2 is chosen for 0.3.

Furthermore, for the qualitative evaluation, we compare our method with [2] and show some single salient object detection results in Figure 9. Generally speaking, our approach provides reasonable results. For the example of the flower in group 1 and the sign boards in group 2, the baseline method finds the part with high contrast, but fails to detect low-contrast salient regions. Our method is basically different from feature-based algorithms and such low-contrast salient regions are successfully obtained.

However for some concave parts of object like the frog in group 2 and the horses in group 3, distance adjustment from concavity context are not powerful enough to make the whole foreground object separable, and our method somehow fails in these images. Meanwhile, the result of our algorithm to some extent depends on the segmentation and clustering algorithms. Figure 10 and 11 illustrate some failure examples of these kinds.

4.2. Multiple Salient Object Detection

Most state-of-the-art salient object detection algorithms provide binary results, which is to say, they regard all the detected regions as one object. However our method is capable of finding multiple salient objects or parts simultaneously by continuously partitioning the foreground subgraph. Based on our study, there is no suitable database and method to evaluate the multiple salient object detection. Hence we only show some results in Figure 12.

4.3. Discussion

Computation Efficiency. In our approach, we do not need to extract low-level image features, and all the computations are done in $O(n)$ time, including the Ncut algorithm [26], hence our method is efficient.

Future Work. Exploiting mid-level perceptual cues in variant computer vision tasks is always an intriguing direction of work. Ren et al. [21] propose to integrate different FGO cues to assign figure/ground labels for object contours using a conditional random field model. Sundberg et al. [27] further utilize some mid-level cues from optical flow to estimate figure/ground relations. In the future, we plan to further study the perceptual cues in the task of FGO.

Moreover, there are many other promising applications of mid-level perceptual cues besides FGO, such as depth estimation [3], edge detection [20], part detection for object parsing and recognition [18], etc. In our future work, we also plan to make use of such mid-level cues in other challenging applications.

5. Conclusion

In this paper, we formulate salient object detection as a figure-ground organization (FGO) problem. We utilize the the cue of convexity (concavity) and design a hierarchical model to add up concavity context on different scales in the hierarchy. A normalized graph-cut algorithm is finally used to segregate foreground salient objects. Experiments show that this spatial context is effective in the task of salient object detection compared with other state-of-the-art feature-based algorithms.

6. Acknowledgement

We appreciate the valuable comments from anonymous reviewers. This work was supported in part by the 973 Program (No.2010CB327906), and the NSF of China (No.60903077, and No.60873178).

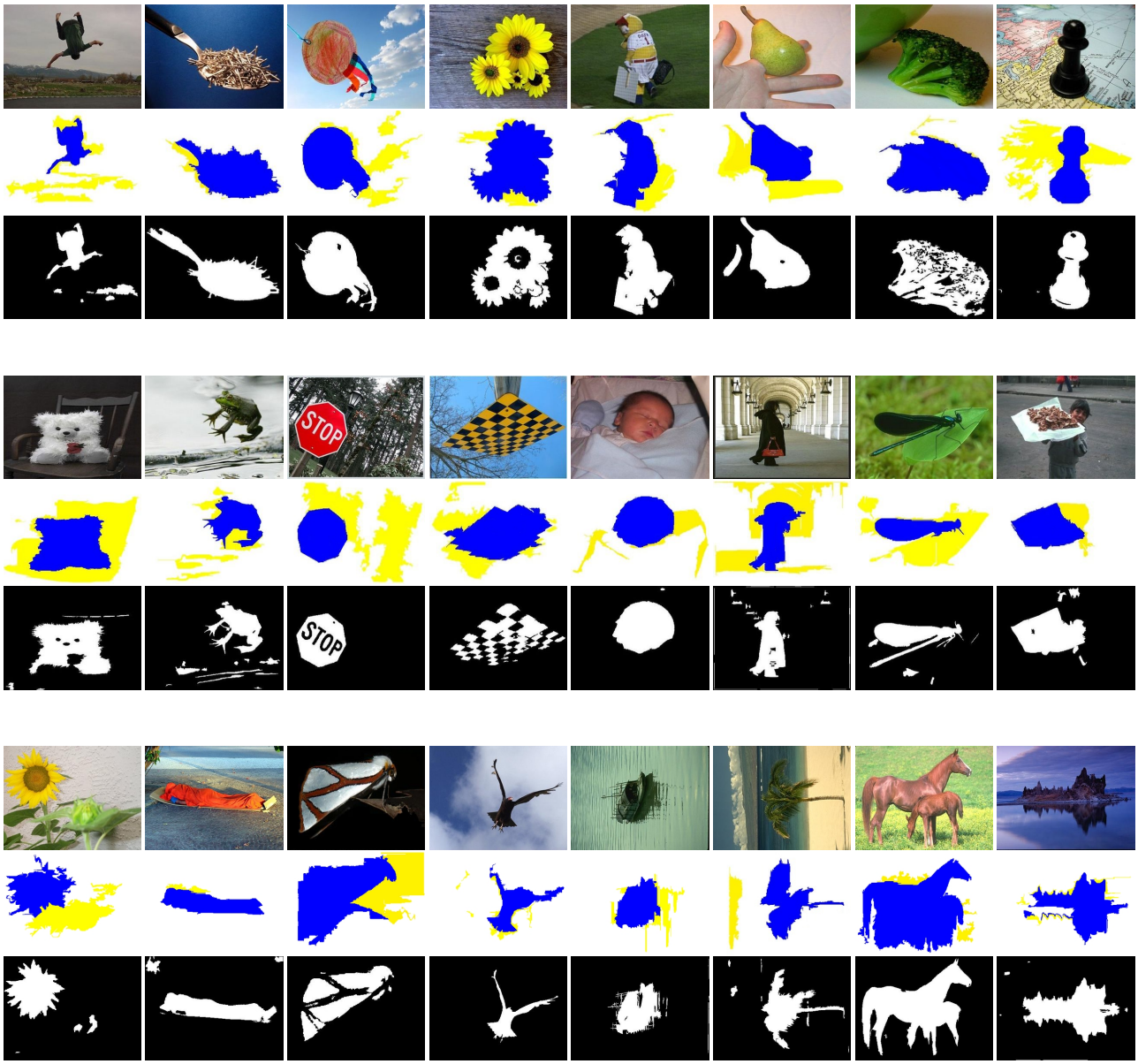


Figure 9. Examples of single salient object detection results. For each group, the first row is input images. The second row is our detection results. The third row is results from [2]. In our results, the blue part is salient object and yellow part represents background objects.

Figure 10. Failure examples caused by pool segmentation results (middle). In these examples the background objects contain too much concavity information and few concave arcs belong to salient objects are detected.

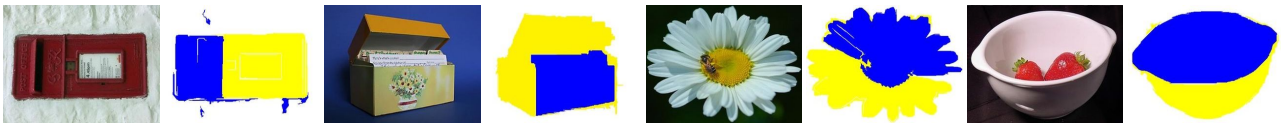


Figure 11. Some other failure examples caused by the Ncut algorithm and hierarchical part of salient objects.

Figure 12. Some successful results for multiple/hierarchical salient object detection.

References

[1] R. Achanta, F. Estrada, P. Wils, and S. Susstrunk. Salient region detection and segmentation. *Computer Vision Systems*, pages 66–75, 2008. 6

[2] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *CVPR*, pages 1597–1604, 2009. 1, 5, 6, 7

[3] J. Burge, C. Fowlkes, and M. Banks. Natural-scene statistics predict how the figure–ground cue of convexity affects human depth perception. *The Journal of Neuroscience*, 30(21):7269, 2010. 6

[4] H. Dorksen-Reiter and I. Debled-Rennesson. Convex and concave parts of digital curves. *Geometric Properties for Incomplete Data*, 31:145–159, 2006. 3

[5] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results. http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html. 2

[6] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59(2):167–181, 2004. 1, 2, 3

[7] C. Fowlkes, D. Martin, and J. Malik. Local figure–ground cues are valid for natural images. *Journal of Vision*, 7(8), 2007. 2

[8] J. Han, K. Ngan, M. Li, and H. Zhang. Unsupervised extraction of visual attention objects in color images. *TCSVT*, 16(1):141–145, 2005. 1

[9] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. In *NIPS*, page 545, 2007. 1, 6

[10] D. Hoiem, A. Efros, and M. Hebert. Recovering surface layout from an image. *IJCV*, 75(1):151–172, 2007. 3

[11] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *CVPR*, pages 1–8. Ieee, 2007. 1, 6

[12] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *TPAMI*, 20(11):1254–1259, 1998. 1, 6

[13] G. Kanizsa and W. Gerbino. Convexity and symmetry in figure-ground organization. *Vision and artifact*, pages 25–32, 1976. 1

[14] O. Le Meur, P. Le Callet, D. Barba, and D. Thoreau. A coherent computational approach to model bottom-up visual attention. *TPAMI*, pages 802–817, 2006. 1

[15] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H. Shum. Learning to detect a salient object. *TPAMI*, 2010. 1, 5

[16] Y. Ma and H. Zhang. Contrast-based image attention analysis by using fuzzy growing. In *ACM Conf. on Multimedia*, pages 374–381, 2003. 1, 6

[17] T. Malisiewicz and A. Efros. Improving spatial support for objects via multiple segmentations. In *BMVC*, 2007. 3, 5

[18] S. Palmer. *Vision science: Photons to phenomenology*. MIT press Cambridge, MA., 1999. 1, 3, 5, 6

[19] M. Peterson. Object recognition processes can and do operate before figure-ground organization. *Current Directions in Psychological Science*, 3(4):105–111, 1994. 1

[20] X. Ren, C. Fowlkes, and J. Malik. Mid-level cues improve boundary detection. *Computer*, 2005. 6

[21] X. Ren, C. Fowlkes, and J. Malik. Cue integration for figure/ground labeling. In *NIPS*, page 1121, 2006. 6

[22] X. Ren, C. Fowlkes, and J. Malik. Figure/ground assignment in natural images. In *ECCV*, pages 614–627, 2006. 2

[23] T. Roussillon, I. Sivignon, and L. Tougne. Robust decomposition of a digital curve into convex and concave parts. In *ICPR*, 2008. 3

[24] E. Rubin. *Visuell wahrgenommene figuren: Studien in psychologischer analyse*. Gyldendalske boghandel, 1921. 1

[25] B. Russell, W. Freeman, A. Efros, J. Sivic, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *CVPR*, pages 1605–1614, 2006. 3

[26] J. Shi and J. Malik. Normalized cuts and image segmentation. *TPAMI*, 22(8):888–905, 2002. 5, 6

[27] P. Sundberg, T. Brox, M. Maire, P. Arbeláez, and J. Malik. Occlusion boundary detection and figure/ground assignment from optical flow. In *CVPR*, 2011. 6

[28] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *ICCV*, 2005. 2

[29] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010. 2

[30] S. Yu and J. Shi. Object-specific figure-ground segregation. In *CVPR*, 2003. 4

[31] Z. Yu. A rule based technique for extraction of visual attention regions based on real-time clustering. *TMM*, 9(4):766–784, 2007. 1

[32] W. Zhang, Q. Wu, G. Wang, and H. Yin. An Adaptive Computational Model for Salient Object Detection. *TMM*, 12(4):300–316, 2010. 1