SALIENCY DETECTION VIA MULTI-CENTER CONVEX HULL PRIOR

Zhi-Jie Wang[†,⊥], *Lizhuang Ma*[¬], *Xiao Lin*[‡], *Hui Zhong*[#]

[†]Sun Yat-Sen University [⊥]Guangdong Key Laboratory of Big Data Analysis & Processing
[¬]Shanghai Jiao Tong University [‡]Shanghai Normal University [#]Shanghai Yingcheng Image Tech. Co., Ltd

ABSTRACT

Saliency detection has been a hot topic in computer vision. Among existing approaches, a representative one is to use the convex hull prior to find the salient object in the image; and there are many variants that are based on the convex hull prior. Most of these works used a single center to construct the convex hull center prior map, while few attention has been made on the use of multiple centers. In this paper, we propose a multi-center convex hull prior based solution for saliency detection. Particularly, our solution also integrates two non-trivial optimizations: one is for obtaining an enhanced *global color distinction prior map*, and another is for refining the preliminary *saliency map*. We experimentally evaluate our solution through comparing against state-of-the-art algorithms. The results demonstrate the effectiveness and superiorities of the proposed solution.

Index Terms— Convex hull; multi-center; saliency detection

1. INTRODUCTION

Visual saliency [1] is one of classical ways to find the regions of interest in the image. In the past decades, abundant efforts have been made on salient object detection, due to its widely applications such as image classification, visual recognition, and text recognition [2, 3, 4, 5, 6, 7]. In the literature, existing saliency detection algorithms can be generally classified into two categories: (i) the top-down approaches, which are task-driven [8, 9, 10, 11, 12, 13]; and (ii) the bottom-up approaches, which are data-driven, and generate the final saliency map by directly simulating the underlying low-level visual attributes [3, 14, 15].

The bottom-up approaches have many branches [14, 15, 16, 17], and one of representative approaches is the *convex hull*-based approach [18, 19]. A major feature of this approach is to approximately locate the foreground seeds via the points of interest, see e.g., [20, 14, 21, 22, 23]. In these literature, most of works utilized a single center to construct the convex hull center prior map, while few efforts have been taken to use multiple centers. One can observe that, in many images there may exist multiple targets/objects, the convex hull generated from these targets could be the polygon with a large *span* in some direction, or the similar cases; see e.g., the first row in Fig. 1. Essentially, for images in which only a singe target/object appears, the above phenomenon could also appear; see the second row in Fig. 1. The single-center convex hull based algorithms in these cases could fail to highlight the salient object effectively (cf., Fig. 1(c)), since it always assigns the high scores to the regions close to the center of the convex hull (cf., Fig. 1(b)).

Inspired by the reasons above, this paper suggests a multi-center convex hull prior-based solution (Section 2). The central contribution of our solution is the construction of the *multi-center prior* (MCP) map (Section 2.1). Besides, our solution also develops two non-trivial optimizations: (i) a more robust *global color distinction prior* (GCDP) map (Section 2.2); and (ii) an improved Bayesian optimization framework (Section 2.3). Our model is easy-to-understand and implement, but without loss of effectiveness. Empirical study on benchmark datasets shows the competitiveness of our proposed model (Section 3).

2. OUR SOLUTION

In a nutshell, our solution works as follows. It first constructs an *MCP map* and an enhanced *GCDP map* respectively. Then, it integrates them, getting a preliminary *saliency map*. It finally employs an improved Bayesian optimization method to obtain the refined *saliency map*.

2.1. Constructing MCP map

Firstly, we use an existing algorithms [23] to construct a convex hull C, and then use *Harris algorithm* [24] to find some points of interest in C. Assume there are N_p points of interest (found by Harris algorithm). Second, we choose a set of k centers *dynamically*. For clarity, we call this algorithm *dynamic k-center algorithm*. Third, we construct the MCP map based on these k centers. In what follows, we focus on the second and third steps.

► *Dynamic k-center algorithm.* For ease of presentation, we first clarify several concepts. Let n_l be the number of edges of C, and let c denote the centroid of C. In addition, for the ith edge l_i, we denote by $dist(l_i, c)$ the distance between the edge l_i and the centroid c. The convex hull centroid radius, denoted by r, is defined as

$$r = \text{argmin}_{i\in[1,n_l]}(dist(l_i, c)) \tag{1}$$

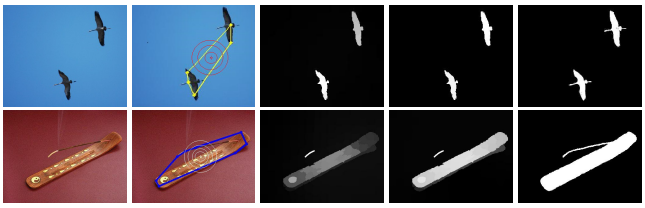
In addition, given a set S_n of n points in C, the coverage ratio of S_n in C, denoted by η, is defined as

$$\eta = \frac{\alpha(C \cap (\bigcup_{i=1}^{n} \odot(i, r+\epsilon)))}{\alpha(C)} \tag{2}$$

where $\alpha(\cdot)$ denotes the area of a geometry, $\odot(i, r+\epsilon)$ denotes a circle whose center and radius are the ith point in S_n and $r+\epsilon$, respectively. Note that, here ϵ is a parameter used to alleviate a too small radius. In our paper, we set ϵ as follows.

$$\epsilon = \text{argmin}_{i,j\in[1,N_p],i\neq j}(dist(PI_i, PI_j)) \tag{3}$$

∗ This work was supported by the NSFC (No. 61472245, 61472453, 61502220, 61775139, U1401256, U1501252, U1611264, U1711261, and U1711262). Email: wangzhij5@mail.sysu.edu.cn, ma-lz@cs.sjtu.edu.cn, lin6008@shnu.edu.cn, zhhzhonghui@gmail.com.



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

Fig. 1. From left to right: (a) input image; (b) explain; (c) saliency result via single-center; (d) our result; (d) ground-truth.

where PI_i and PI_j denote two different points of interest.

Similarly, the overlap region size, denoted by s, is defined as

$$s = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \alpha(\odot(i, r+\epsilon) \cap \odot(j, r+\epsilon)) \tag{4}$$

where $\cap$ denotes the intersection set of two circles. Based on s, we define the overlap degree, denoted by ι, as follows.

$$\iota = \frac{s}{\sum_{i=1}^{n} \alpha(\odot(i, r+\epsilon))} = \frac{s}{n \times \alpha(\odot(i, r+\epsilon))} \tag{5}$$

Let τ_η and τ_ι be two thresholds used for η and ι, respectively. The algorithm works as follows. Firstly, it uses the set of N_p points of interest as the initial centers. It computes η and ι based on the above equations, and sets $\tau_\eta = d * \eta$, where $d \in [0, 1]$. In our paper, the parameters d and τ_ι are empirically chosen, and set to 0.85 and 0.4, respectively. Then, it checks whether ι is less than τ_ι, where $\tau_\iota \in [0, 1]$. If so, the algorithm terminates and returns these N_p points as the centers. Otherwise, it uses *k-means algorithm* [25] to group these N_p points into $\lfloor \frac{N_p}{2} \rfloor$ clusters, and uses the centers of these $\lfloor \frac{N_p}{2} \rfloor$ clusters to compute "new" η and ι. This iteration terminates until one of the following two conditions is satisfied: (i) $\iota < \tau_\iota$, or (ii) $\eta < \tau_\eta$. Assume that our algorithm terminates after i times iterations, it shall return $\lfloor \frac{N_p}{2^i} \rfloor$ centers. In other words, $k = \lfloor \frac{N_p}{2^i} \rfloor$. Note that, the intuition behind our algorithm is to select appropriate (i.e., k) points, based on points of interest, such that these k points can cover C as much as possible (note: each point is associated with a circle), while the overlap degree of these circles should be small.

► *MCP map construction phase.* Observe that, the k centers obtained could be not uniformly distributed in C. To alleviate this issue, we give different weights to different centers (when we construct the MCP map). The weights are computed as follows. We first construct a $k \times k$ matrix $\mathbb{M}$, based on the distance of each pair of centers. That is, $\mathbb{M} = [dist(i, j)]_{k \times k}$, where $i, j \in [1, k]$. Then, for each column in $\mathbb{M}$, we accumulate all the values, getting a vector $\mathbb{V}$. That is,

$$\mathbb{V} = \{\sum_{i=1}^{k} dist(i, 1), \sum_{i=1}^{k} dist(i, 2), ..., \sum_{i=1}^{k} dist(i, k)\} \tag{6}$$

We normalize the vector $\mathbb{V}$, and finally obtain the MCP map by processing each each superpixel sp_i as follows.

$$S_i^{fg} = \frac{1}{k} \sum_{j=1}^{k} v_j \times e^{(-\frac{(x_i - x_j)^2}{2\delta_x^2} - \frac{(y_i - y_j)^2}{2\delta_y^2})} \tag{7}$$

where v_j denotes the jth item in $\mathbb{V}$; x_i (resp., y_i) denotes the mean horizontal (resp., vertical) coordinates of sp_i; δ_x (resp., δ_y) denotes the horizontal (resp., vertical) variance; in our paper we set $\delta_x = \delta_y = 0.5$, unless stated otherwise.

2.2. Obtaining a more robust GCDP map

This section shows how to construct the enhanced *GCDP map*. Generally speaking, our method can be viewed as a fusion of two ideas: (i) using separately each side of image boundary to construct enhanced GCDP maps; and (ii) utilizing fully the convex hull (obtained before) to further optimize the GCDP maps.

► *Enhanced GCDP map construction phase.* Existing methods usually group the superpixels in all boundaries into three clusters and then construct GCDP maps [26]. Instead, we construct four GCDP maps based on four boundaries. The intuition behind this idea is that, the background often presents the local or global consistence with one or some of these boundaries. Without loss of generality, assume there are N superpixels in the image, and the number of superpixels in the mth boundary is n_m ($m \in [1, 4]$), our method constructs the mth GCDP map, denoted by $\mathbb{G}_m$, as follows.

$$\begin{cases} \mathbb{G}_m = [S_{m,i}]_{1 \times N} & \text{(8a)} \\ S_{m,i} = \frac{1}{n_m} \sum_{j=1}^{n_m} \frac{1}{w(i,j) + \beta} & \text{(8b)} \end{cases}$$

where the element $S_{m,i}$ denotes the saliency value of superpixel i in the mth GCDP map; $w(i, j) = exp(-\frac{\|c_i, c_j\|}{2\delta_1^2})$, it measures the color similarity of the ith and jth superpixels. In our paper the balance parameters $\delta_1 = 0.2$ and $\beta = 10$, respectively. Our method above is inspired by [27], but different from theirs, since they use Manifold ranking to construct the side-specific map, while we use Eqs. 8a and 8b to construct the GCDP map.

► *Further optimize the GCDP map.* It is not hard to see that the GCDP map obtained above could restrain the saliency value of some superpixel regions whose colors are similar to the boundaries. To address this issue, we further optimize the above map by fully utilizing the convex hull obtained previously. This optimization can be achieved by revising Equation 8b. Specifically, for each superpixel sp_i we first compute a value, denoted by D_i, as follows.

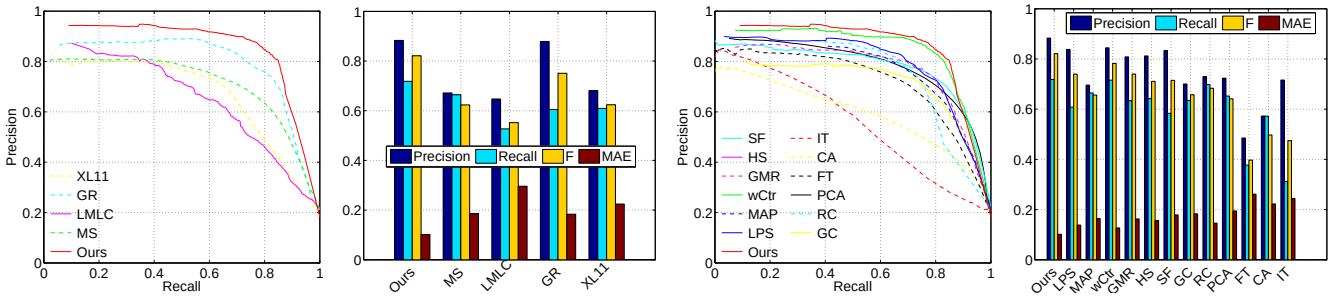
$$D_i = \begin{cases} \varphi_1 \sum_{j=1}^{n_m} (1 - \frac{1}{w(i,j) + \beta}) & sp_i \in R_F \\ \varphi_2 \sum_{j=1}^{n_m} \frac{1}{w(i,j) + \beta} & sp_i \in R_B \end{cases} \tag{9}$$

where φ_1 and φ_2 are two weight factors, which are set to 0.8 and 0.2; R_F (resp., R_B) denotes the regions inside (reps., outside of) C. By integrating D_i into Equation 8b, obtaining the following.

$$S_{m,i} = \frac{1}{n_m} (\sum_{j=1}^{n_m} \frac{1}{e^{(-\frac{\|c_i, c_j\|}{2\delta_1^2})} + \beta} + D_i) \tag{10}$$

Finally, we merge all the four GCDP maps, obtaining the final GCDP map $\mathbb{G}$.

$$\begin{cases} \mathbb{G} = [S_i^{bg}]_{1 \times N} & \text{(11a)} \\ S_i^{bg} = \prod_{m=1}^{4} S_{m,i}, \quad i \in [1, N] & \text{(11b)} \end{cases}$$



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

Fig. 2. Comparison results: (a-b) *convex hull*-based methods; (c-d) other classic and/or state-of-the-art methods.

where S_i^{bg} denotes the saliency value of the ith superpixel.

Remark that, the method in [28] can be immediately used to integrate our MCP and GCDP maps. Specifically, for each superpixel sp_i, let S_i denote the saliency value after integration. It is computed as $S_i = S_i^{fg} \times (1 - e^{(-\lambda S_i^{bg})})$, where λ is a balance factor. Following [28], in our experiments λ is set to 6, unless stated otherwise.

2.3. Refining the saliency map

While the integration method mentioned before can work correctly, the edges of the salient object could be not well preserved [20]. Previous works (see e.g., [20, 14]) have attempted to improve the saliency map via *Bayesian optimization*. This optimization addresses the issue above, and suppresses favourably the background noise outside of the saliency region. Yet, it ignores another issue: the saliency value of superpixel regions could be also restrained when these superpixel regions are similar to the background. To fix it, we employ an improved Bayesian optimization framework to refine the initial saliency map. The rationale behind our method is to assign larger weights for superpixels in C even if these superpixel regions are similar to the background. Specifically, we do as follows.

Firstly, for each superpixel sp_i, we define the superpixel region weight w_i as follows.

$$w_i = \begin{cases} e^{-g_i}, & if \quad sp_i \in R_F \\ e^{-(g_i+u)}, & if \quad sp_i \in R_B \end{cases} \tag{12}$$

where $u \in (0, 2)$ is a parameter used to balance the size of the exponent item, it is empirically set to 1; and g_i is computed as

$$\begin{aligned} g_i = & \frac{1}{N_1} \sum_{j_1=1}^{N_1} Norm(\|c_i, c_{j_1}\|) \\ & + (1 - \frac{1}{N_2} \sum_{j_2=1}^{N_2} Norm(\|c_i, c_{j_2}\|)) \end{aligned} \tag{13}$$

where N_1 (resp., N_2) is the number of superpixels in R_F (resp., R_B), and $Norm(\|c_i, c_{j_1}\|)$ is obtained as follows (notice: $Norm(\|c_i, c_{j_2}\|)$ is obtained similarly).

$$Norm(\|c_i, c_{j_1}\|) = \frac{\|c_i, c_{j_1}\| - \|c_i, c_{j_1}\|_{min}}{\|c_i, c_{j_1}\|_{max} - \|c_i, c_{j_1}\|_{min}} \tag{14}$$

where $\|c_i, c_{j_1}\|_{max}$ refers to $\text{argmax}_{j_1 \in [1,N_1]}(\|c_i, c_{j_1}\|)$, and $\|c_i, c_{j_1}\|_{min}$ refers to $\text{argmin}_{j_1 \in [1,N_1]}(\|c_i, c_{j_1}\|)$.

Second, for each superpixel region sp_i we compute the "weighted" observation likelihood. That is,

$$\begin{cases} p_w(sp_i|sal) = \dfrac{w_i}{\sum_{sp_j \in sal} w_j} & \text{(15a)} \\ p_w(sp_i|bg) = \dfrac{w_i}{\sum_{sp_j \in bg} w_j} & \text{(15b)} \end{cases}$$

where sal (resp., bg) denotes the convex hull region R_F (resp., R_B); $p_w(s_i|sal)$ (resp., $p_w(s_i|bg)$) refers to the weighted observation likelihood of the superpixel sp_i in sal (resp., bg).

Then, for each pixel v in the image, we compute the observation likelihood as follows.

$$\begin{cases} p(v|sal) = \sum_{sp_i \in sal} p_w(sp_i|sal) p(v|sp_i) & \text{(16a)} \\ p(v|bg) = \sum_{sp_i \in bg} p_w(sp_i|bg) p(v|sp_i) & \text{(16b)} \end{cases}$$

where $p(v|sp_i)$ is the observation likelihood of pixel v in the superpixel sp_i, $p(v|sal)$ (resp., $p(v|bg)$) is the observation likelihood of the pixel v in sal (resp., bg). The final step is the same as that in [20, 14], it constructs the final saliency map by Bayesian formula as follows: $p(sal|v) = \frac{p(sal)p(v|sal)}{p(sal)p(v|sal)+p(bg)p(v|bg)}$.

3. EXPERIMENTAL EVALUATION

In our experiments, we use three evaluation metrics. (i) The *precision-recall* (P-R) curve. We follow prior works [14, 26] to obtain it. (ii) The F-measure, denoted by F_m. It is computed as $F_m = \frac{(1+\varphi^2) \cdot p_v \cdot r_v}{\varphi^2 \cdot p_v + r_v}$, where p_v and r_v denote the precision and recall values respectively, and $\varphi^2 = 0.3$. (iii) The *mean absolute error* (MAE), denoted by ξ. It is computed as $\xi = \frac{1}{n_a} \sum_{i=1}^{n_a} |\mathsf{S}(p_i) - \mathsf{G}(p_i)|$, where n_a denotes the number of all pixels in the image, $\mathsf{S}(p_i)$ and $\mathsf{G}(p_i)$ denote the information of the ith pixel from the saliency map and from the ground true, respectively. We compare our algorithm with two sets of algorithms: (i) various *convex hull*-based algorithms including XL11 [20], GR [22], LMLC [14] and MS [21]; and (ii) other classical or state-of-the-art methods including MAP [29], wCtr [3], SF [30], PCA [17], IT [1], RC [16], HS [15], GC [5], FT [31], CA [32], GMR [27], and LPS [33]. We have evaluated our solution based on several well-known datasets, including SED2 [7], *MSRA-5000* [12], *ASD* [34], *ECSSD* [15], *THUR* [35]; our solution performs well for all these datasets. Due to space limit, we cover the quantitative comparison on SED2 as a sample (cf., Fig. 2).

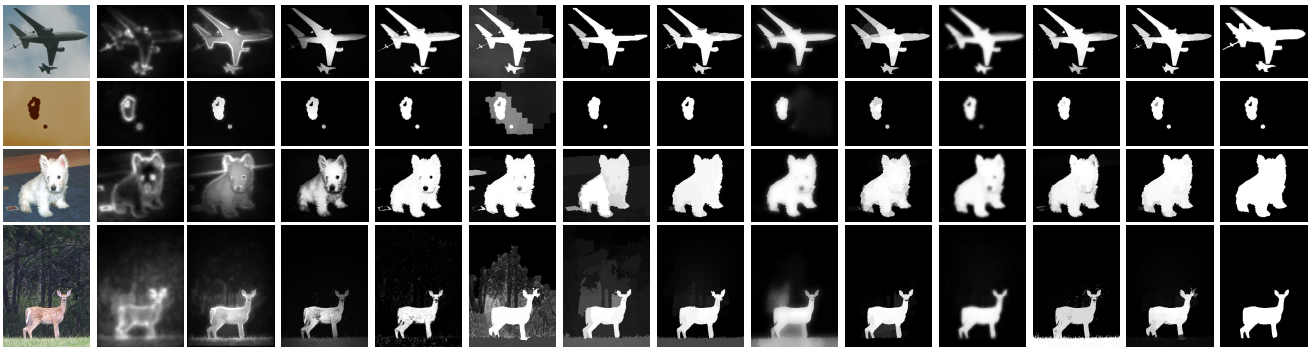


Fig. 3. The saliency maps obtained based on different methods: (a) input images; (b) CA; (c) PCA; (d) SF; (e) GC; (f) LMLC; (g) HS; (h) GMR; (i) MS; (j) wCtr; (k) MAP; (l) LPS; (m) Ours; and (n) GT (ground-truth).

From Fig. 2(a) we can see that, the P-R curve of our method dominates the ones of other convex hull-based algorithms, demonstrating the effectiveness of our solution. Furthermore, from Fig. 2(b) one can see that our MAE (resp., F-measure) value is the smallest (resp., largest) one among all these convex hull-based algorithms. This further verifies its effectiveness. On the other hand, Figs. 2(c) and 2(d) show the comparative results of our algorithm against other algorithms. One can see that the curve of our algorithm is above the ones of all these competitors; in addition, regarding the F-measure and/or MAE, our algorithm also performs better than these competitors. These evidences show that our algorithm is competitive, compared with other types of saliency detection algorithms.

Fig. 3 shows the qualitative comparison, i.e., the saliency maps generated by our algorithm and the competitors. We can see that, for images containing multiple objects (cf., Rows 1 and 2), some competitors can not detect all salient objects, while our method can achieve this; moreover, it highlights all salient objects and restrains background noises favourably. These demonstrate the effectiveness and robust of our model. On the other hand, for images containing a single object (cf., Rows 3 and 4), our algorithm also performs well. This further verifies the superiorities of our model.

Besides, we also examine the effectiveness of the proposed techniques, respectively. Specifically, three baselines are used: baseline 1 — without using the MCP map; baseline 2 — without using the enhanced GCDP map; baseline 3 — without using the improved Bayesian framework. Fig. 4 covers the comparison results. The first row compares baseline 1 and our solution. We can see that the baseline highlights some parts of the saliency region, while other parts in the saliency region are not well highlighted. In contrast, our solution highlights the saliency region uniformly, compared with the baseline. Moreover, one can observe that, for the baseline, some background regions that are near to the center of the "highlighted" saliency region are also highlighted. Yet, our method overcomes this limitation. This essentially demonstrates the effectiveness of our MCP map that assigns different weights for different centers, avoiding over-highlighting some parts. The second row compares baseline 2 and our solution. It can be seen that, for the saliency maps generated by the baseline, some parts in the saliency region are still dark. In contrast, our model can favourably highlight these parts, demonstrating the effectiveness of the enhanced GCDP map. The third row reports the saliency result of baseline 3. Compared with the saliency map generated by the baseline, the map generated by our model further suppresses the noises located in the saliency region. In addition, it also suppresses the background noises. This implies that the improved Bayesian optimization framework can bring us an extra benefit — further suppressing background noises.

Table1 compares the running time on the ASD dataset [34], where "M" means "Matlab", and "C" means "C/C++". All tests are conducted on a PC with 3.50 GHz CPU and 32 GB RAM. It can be seen that our algorithm consumes relatively less time (i.e., no more than 1 *sec*). Combining the superiorities validated in previous paragraphs, on the whole our model is competitive.

4. CONCLUSIONS

In this paper, we proposed a new convex hull-based prior saliency detection model. Our model is easy-to-understand and implement but without loss of effectiveness. The central idea of our model is to use multiple centers to construct the prior map. In addition, our model also incorporates two non-trivial optimizations, which are developed for generating an enhanced GCDP map and for refining the preliminary saliency map, respectively. We verified the superiorities and effectiveness of our solution through extensive experiments. In the future, we would like to adapt our solution and other techniques to specific applications such as visual recognition.

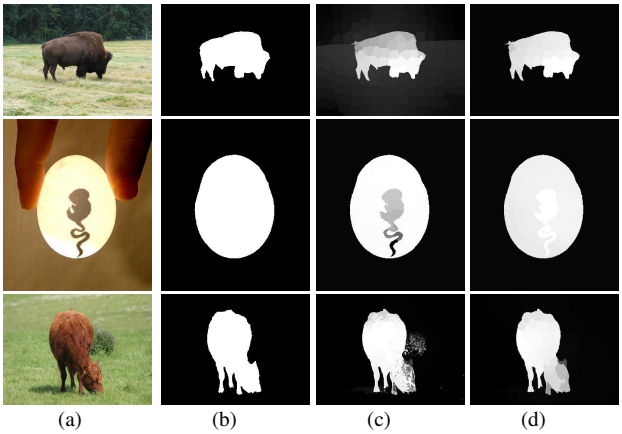


Fig. 4. (a) input image; (b) ground-truth; (c) baselines; (d) our solution.

Table 1. Average running time (second *per* image)

Method	CA	PCA	HS	GMR	MS	LPS	Ours
Time	32.343	3.258	0.407	0.411	2.653	1.076	0.822
Code	M+C	M+C	C	M	M	M	M+C

5. REFERENCES

[1] L. Itti, C. Koch, E. Niebur, *et al.*, "A model of saliency-based visual attention for rapid scene analysis," *TPAMI*, vol. 20, no. 11, pp. 1254–1259, 1998.

[2] K. Fu, I. Y. H. Gu, and J. Yang, "Learning full-range affinity for diffusion-based saliency detection," in *ICASSP*, pp. 1926–1930, 2016.

[3] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *CVPR*, pp. 2814–2821, 2014.

[4] J. Zhang and S. Sclaroff, "Exploiting surroundedness for saliency detection: A boolean map approach," *TPAMI*, vol. 38, no. 5, pp. 889–902, 2016.

[5] M.-M. Cheng, J. Warrell, W.-Y. Lin, S. Zheng, V. Vineet, and N. Crook, "Efficient salient region detection with soft image abstraction," in *ICCV*, pp. 1529–1536, 2013.

[6] Z. Wu, L. Su, Q. Huang, B. Wu, J. Li, and G. Li, "Video saliency prediction with optimized optical flow and gravity center bias," in *ICME*, pp. 1–6, 2016.

[7] S. Alpert, M. Galun, A. Brandt, and R. Basri, "Image segmentation by probabilistic bottom-up aggregation and cue integration," *TPAMI*, vol. 34, no. 2, pp. 315–327, 2012.

[8] J. Kuen, Z. Wang, and G. Wang, "Recurrent attentional networks for saliency detection," in *CVPR*, pp. 3668–3677, 2016.

[9] S. He and R. W. H. Lau, "Exemplar-driven top-down saliency detection via deep association," in *CVPR*, pp. 5723–5732, 2016.

[10] P. Jiang, N. Vasconcelos, and J. Peng, "Generic promotion of diffusion-based salient object detection," in *ICCV*, pp. 217–225, 2015.

[11] N. Liu and J. Han, "Dhsnet: Deep hierarchical saliency network for salient object detection," in *CVPR*, pp. 678–686, 2016.

[12] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," *TPAMI*, vol. 33, no. 2, pp. 353–367, 2011.

[13] A. Borji, D. N. Sihite, and L. Itti, "Probabilistic learning of task-specific visual attention," in *CVPR*, pp. 470–477, 2012.

[14] Y. Xie, H. Lu, and M.-H. Yang, "Bayesian saliency via low and mid level cues," *TIP*, vol. 22, no. 5, pp. 1689–1698, 2013.

[15] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *CVPR*, pp. 1155–1162, 2013.

[16] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, "Global contrast based salient region detection," *TPAMI*, vol. 37, no. 3, pp. 569–582, 2015.

[17] R. Margolin, A. Tal, and L. Zelnik-Manor, "What makes a patch distinct?," in *CVPR*, pp. 1139–1146, 2013.

[18] R. Liu, J. Cao, Z. Lin, and S. Shan, "Adaptive partial differential equation learning for visual saliency detection," in *CVPR*, pp. 3866–3873, 2014.

[19] C. Gong, D. Tao, W. Liu, S. J. Maybank, M. Fang, K. Fu, and J. Yang, "Saliency propagation from simple to difficult," in *CVPR*, pp. 2531–2539, 2015.

[20] Y. Xie and H. Lu, "Visual saliency detection based on bayesian model," in *ICIP*, pp. 645–648, 2011.

[21] N. Tong, H. Lu, L. Zhang, and X. Ruan, "Saliency detection with multi-scale superpixels," *IEEE Signal Processing Letters*, vol. 21, no. 9, pp. 1035–1039, 2014.

[22] C. Yang, L. Zhang, and H. Lu, "Graph-regularized saliency detection with convex-hull-based center prior," *IEEE Signal Processing Letters*, vol. 20, no. 7, pp. 637–640, 2013.

[23] H. Zhu, B. Sheng, X. Lin, Y. Hao, and L. Ma, "Foreground object sensing for saliency detection," in *ICMR*, pp. 111–118, 2016.

[24] J. Van de Weijer, T. Gevers, and A. D. Bagdanov, "Boosting color saliency in image feature detection," *TPAMI*, vol. 28, no. 1, pp. 150–156, 2006.

[25] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques, Third Edition*. Massachusetts: Morgan Kaufmann, 2011.

[26] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata," in *CVPR*, pp. 110–119, 2015.

[27] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *CVPR*, pp. 3166–3173, 2013.

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

[29] J. Sun, H. Lu, and X. Liu, "Saliency region detection based on markov absorption probabilities," *TIP*, vol. 24, no. 5, pp. 1639–1649, 2015.

[30] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *CVPR*, pp. 733–740, 2012.

[31] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *CVPR*, pp. 1597–1604, 2009.

[32] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *TPAMI*, vol. 34, no. 10, pp. 1915–1926, 2012.

[33] H. Li, H. Lu, Z. Lin, X. Shen, and B. Price, "Inner and inter label propagation: salient object detection in the wild," *TIP*, vol. 24, no. 10, pp. 3176–3186, 2015.

[34] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *TPAMI*, vol. 34, no. 11, pp. 2274–2282, 2012.

[35] M.-M. Cheng, N. J. Mitra, X. Huang, and S.-M. Hu, "Salientshape: Group saliency in image collections," *The Visual Computer*, vol. 30, no. 4, pp. 443–453, 2014.