Neurocomputing 312 (2018) 239–250

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Saliency detection via affinity graph learning and weighted manifold ranking☆

Xinzhong Zhu[a,b,1], Chang Tang[c,1], Pichao Wang[d], Huiying Xu[b], Minhui Wang[e,*], Jiajia Chen[e,*], Jie Tian[a]

[a] School of Electronic Engineering, XIDIAN University, Xi'an 710071, PR China
[b] College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, Jinhua 321004, PR China
[c] School of Computer Science, China University of Geosciences, Wuhan 430074, PR China
[d] School of Computing and Information Technology, University of Wollongong, NSW 2522, Australia
[e] Department of Pharmacy, Huai'an Second People's Hospital Affiliated to Xuzhou Medical College, Huai'an, 223002, China

ARTICLE INFO

Article history:
Received 15 August 2017
Revised 27 April 2018
Accepted 28 May 2018
Available online 15 June 2018

Communicated by Li Bing

Keywords:
Saliency detection
Manifold ranking
Affinity graph learning

ABSTRACT

Graph-based saliency detection approaches have gained great popularity due to the simplicity and efficiency of graph algorithms. In these approaches, the saliency values of image elements are ranked by the similarity of image elements with foreground or background cues via graph-based ranking. However, in previous methods, the similarity between any two image elements on the affinity graph is computed by manually set functions which are sensitive to function parameters, and the constructed graph may not reveal the essentially relevance between feature vectors extracted from different image elements. In addition, during the saliency ranking process, all the initial labels contribute equally to the ranking function while the global saliency confidence of each image element is not taken into consideration. In order to address these two issues, we propose a bottom-up saliency detection approach by affinity graph learning and weighted manifold ranking. An unsupervised learning approach is introduced to learn the affinity graph based on image data self-representation. By setting image boundary superpixels as background seeds, the global saliency confidence prior implied in the affinity matrix is utilized to weight the saliency ranking. In such a manner, the superpixels with higher saliency confidences will be assigned higher saliency values in the final saliency map and the background superpixels can be efficiently suppressed. Comprehensive evaluations on three challenge datasets indicate that our algorithm universally surpasses other unsupervised graph based saliency detection methods.

1. Introduction

Human beings can rapidly identify the visually distinctive parts in a given scene, but how to make computer accomplish this task in a quick and unsupervised manner is a challenge problem, which is also called "visual saliency detection" in computer vision field. Visual saliency has been a fundamental research problem in neuroscience, psychology and vision perception for a long time. Salient object detection, as an important and useful branch of visual saliency detection, aims to locate and segment the most informative foreground objects from a scene. It has been a booming research topic in the last one and a half decades due to its wide range of applications in computer vision, such as object detection and recognition [1–3], image classification [4] and retrieval [5–7], object co-segmentation [8,9] and content-based image editing [10–13].

Over the past decades, many saliency detection models have been proposed to compute the saliency map of a given image and detect the salient objects. These models can be mainly categorized into two classes: top-down and bottom-up. Top-down models [14,15] are task-driven and usually exploit high-level human perceptual knowledge, such as context, semantic and background priors, to guide the saliency detection. However, the generalization and scalability of top-down models are limited due to the high diversity of object and task types. Bottom-up methods [16–19] are stimulus-driven and usually exploit low-level image attributes such as color, gradient, edges, and boundaries to construct saliency maps. Compared to traditional methods with low-level hand-crafted features, deep neural networks have also achieved

☆ Fully documented templates are available in the elsarticle package on CTAN.

* Corresponding author.
E-mail addresses: minhwang@163.com (M. Wang), jjiachen@outlook.com (J. Chen).

[1] indicates equal contribution as first author.

https://doi.org/10.1016/j.neucom.2018.05.106
0925-2312/

state-of-the-art results in saliency detection [20–25]. The success stems from the expressibility and capacity of deep architectures that facilitates learning complex high-level features and models to account for interacted relationships directly from training examples. Recent surveys on salient object detection models can be found in [26–28].

In recent years, due to the simplicity and efficiency of graph algorithms, more and more graph-based saliency detection approaches have been proposed and achieved great success. Harel et al. [29] proposed the graph based visual saliency (GBVS), a graph-based bottom-up saliency model with dissimilarity measurements to extract saliency information. Random walks model [30] has been exploited in an automatic salient-region-extraction method to effectively detect the rough location of the most salient object in an image. Chang et al. [31] introduced a computational framework by constructing a graphical model to fuse objectness and regional saliency. Yang et al. [32,33] utilized the four boundaries of the input image as background queries, the saliency detection is converted to a manifold ranking process on a graph. By taking the image details and region-based estimations into account, Li et al. [34] proposed a regularized random walks ranking to formulate pixel-wised saliency maps from the superpixel-based background and foreground saliency estimations. Wang et al. [35] proposed a novel graph model which can effectively capture local and global saliency cues, the saliency detection is also accomplished by manifold ranking.

Most of above mentioned graph based saliency detection models can be regarded as a ranking or label propagation problem [36], and these algorithms universally outperforms most of the state-of-the-art saliency detection methods and are more computationally efficient. However, the affinity graph (i.e., similarity matrix of feature vectors) used in existing graph based saliency detection methods are computed by manually set functions which are sensitive to function parameters, and this graph may cannot reveal the essentially relevance between feature vectors extracted from different image elements. In addition, during the saliency ranking process, the global saliency confidence of each image element is not taken into consideration.

In this paper, we propose a bottom-up saliency detection approach via affinity graph learning and weighted manifold ranking. Different to previous graph based saliency detection methods, the affinity graph is learned in a unsupervised manner in our approach, but not computed by manually set functions. By using the learned affinity graph, the saliency map is obtained by a two stage weighted manifold ranking. Specifically, we first exploit boundary prior for selecting background ranking seeds to perform an initial background query. Then the initial saliency map is further used to generate foreground ranking seeds to perform salient foreground query to obtain the final saliency map. During the ranking process, the global saliency confidence implied in the affinity graph is used to weight the ranking function, which makes those image regions with higher saliency confidence being allocated with higher saliency scores in the final results. The contributions of this paper can be summarized as follows:

1. We propose an automatic affinity graph learning model for graph-based saliency detection, and a recursive algorithm is developed to solve the model.
2. The global saliency confidence implied in the affinity graph is used to weight the saliency ranking process, which makes the background and foreground regions more separable.
3. Comprehensive evaluations on three challenge datasets are conducted to verify the efficacy of our saliency detection method.

The remainder of this paper is organized as follows. Section 2 describes some related works. Section 3 introduces the proposed method. Experimental results on three datasets are provided in Section 4. Section 5 concludes the paper.

2. Related works

The core of our work is composed of two parts: affinity graph learning and saliency detection by manifold ranking. In the following sections, we will describe some related works about affinity graph learning and manifold ranking.

2.1. Affinity graph learning

Affinity graph is used in many computer vision tasks such as image segmentation [37,38], classification [39,40] and image data clustering [41]. The key issue in graph based applications is to construct a "good" affinity matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$, in which each element $\mathbf{A}_{ij}$ (a.k.a., edge weight) reflects the similarity between data points $\mathbf{x}_i$ and $\mathbf{x}_j$. Ideally, the affinity should be 1 if they are from the same group[2], 0 otherwise. In traditional methods, the most intuitive way to conduct the data affinity matrix is directly computing distances on the raw data (e.g., k-nearest neighbor [42] or ε-neighborhoods using cosine or heat kernel distances). It should be noticed that the parameter setting in these two kinds of methods will heavily influence the final task performance. Nie et al. [43] developed a more sophisticated method to learn the affinity matrix by adaptively assigning neighbors for each data point based on the local connectivity. However, the affinity matrix constructed on the raw data is unable to well reveal the intrinsic structure of data points. Inspired by the data representation theory [44–46], a huge number of research which exploits the relationship of data representations has been proposed. Sparse representation is a typical and widely used technique which assumes that a complex data can be represented by some data bases. Qiao et al. [47] constructed an L_1 graph with weighted edges using the coefficients of the sparse coding and it is integrated into a locality preserving projection method for human face recognition. The coefficients of sparse representation are also used to construct the graph for semi-supervised classification [48] and multi-label classification [49]. Zhang [50] proposed a novel non-negative low-rank and sparse (NNLRS) graph for semi-supervised learning. The weights of edges in the graph are obtained by seeking a non-negative low-rank and sparse matrix that represents each data sample as a linear combination of others. Different from the sparse representation, Locality constrained Linear Coding (LLC) [51] and collaborative neighborhood representation (CNR) [52] are other two algorithms for data representation. Based on CNR and LLC, Dornaika et al. [53] proposed a graph construction method named weighted regularized least square and a two phase method for object categorization. Sparse graphs such as adopting L_1 regularization can provide sparse graphs which have been proved to be very powerful in many real learning problems. However, their computational cost is very expensive. Moreover, it is not very clear if sparse graphs take into account data locality. In this paper, we propose a new unsupervised graph affinity learning method for saliency detection based on self-representation of image regions, and the Laplacian regularization is used to preserve the local smoothness between image regions. To the best of our knowledge, our work is the first attempt to learn the affinity matrix for ranking based saliency detection in a totally unsupervised way.

[2] We use "group" as a general expression here. For specific problems, the group means cluster for clustering task, image region for segmentation, class for classification.

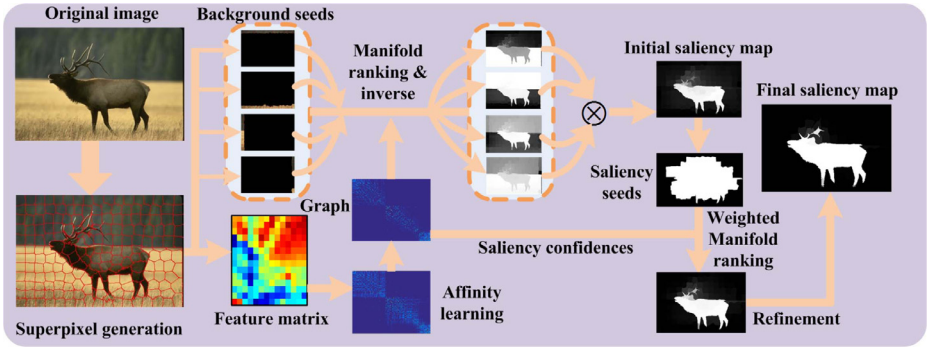


Fig. 1. Pipeline of our proposed saliency detection approach.

2.2. Manifold ranking

Manifold ranking (MR) is initially used for pattern classification [54], and it exploits the intrinsic manifold structure of data for graph labelling. Given a dataset $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_l, \mathbf{x}_{l+1}, \ldots \mathbf{x}_n\} \in \mathbb{R}^{d\times n}$, where d is the dimension of each data, and n is the total number of data points. Part of data points are labelled as queries and the rest need to be ranked according to their relevances to the queries. This can be described by an indicator vector $\mathbf{y} = [y_1, \ldots, y_n]^T$, in which $y_i = 1$ if $\mathbf{x}_i$ is a query, and $y_i = 0$ otherwise. The aim of manifold ranking is to find a ranking function $\mathcal{F} : \mathbf{X} \to \mathbb{R}^n$ which assigns a ranking value f_i to each data point $\mathbf{x}_i$, and it can be written in a vector form $\mathbf{f} = [f_1, \ldots, f_n]^T$. Next, a graph $G = (V, E)$ with nodes V and edges E is established on the dataset, where the nodes V correspond to the data points of $\mathbf{X}$, and the edges E are weighted by an affinity matrix $\mathbf{A} = \left[a_{i,j}\right]_{n\times n}$, the degree matrix of G is $\mathbf{D} = diag\{d_{11}, \ldots, d_{nn}\}$, where $d_{ii} = \sum_{j=1}^{n} a_{ij}$. Then, the optimal ranking solution of queries are obtained by solving the following optimization problem:

$$\hat{\mathcal{F}} = \arg\min_{f} \frac{1}{2}\left(\sum_{i,j=1}^{n} a_{ij}\left\|\frac{f_i}{\sqrt{d_{ii}}} - \frac{f_j}{\sqrt{d_{jj}}}\right\|^2 + \mu\sum_{i=1}^{n}\|f_i - y_i\|^2\right), \quad (1)$$

where the first term in Eq. (1) is the smoothness constraint which forces a good ranking function that should not change too much between nearby points, and the second term is the fitting constraint which implies that the ranking function should not differ too much from the initial query assignment. The parameter μ controls the balance of the two constraints. In the PageRank [55] and spectral clustering algorithms [41], a similar concept is also used. The minimum solution of (1) is computed by setting the derivative of the above function respect to $\mathcal{F}$ to be zero. In previous works [32,34,35], MR is used for saliency detection and achieves great success. In [32,35], MR is directly used in their work and the results outperform most of the state-of-the-art saliency detection methods in terms of accuracy and computationally efficient. Li et al. [34] uses MR to get a coarse saliency map, then the regularised random walking is used to restrict the Dirichlet integral to be as close to the prior saliency distribution for final saliency detection.

Our work is most related to saliency detection based on manifold ranking [32,33] and affinity learning based diffusion [56]. In [32,33], the image was represented as a close-loop graph with superpixels as nodes. These nodes were ranked based on the similarity to background and foreground queries, based on affinity matrices. Saliency detection was carried out in a two-stage scheme to extract background regions and foreground salient objects efficiently. In [56], the authors constructed an initial local neighbour graph by using manually set function, then they learnt an affinity graph by using graph-based semi-supervised learning. With the learned affinity graph, they firstly used four image borders as seeds but removed boundary nodes that are likely to belong to boundary-cropping objects. Next, they used the proposed affinity learning based diffusion to perform diffusion on the seeds separately, and reverse each map to suppress the background and highlight potential objects. Finally, four border-specified maps were superpixel-wisely multiplied to obtain the final saliency map. Compared to previous works, there are new contributions in our work, including: (1) Instead of using the fixed graph pre-computed by manually set functions (i.e., the Gaussian kernel function) as in [33], we propose an automatic affinity graph learning approach to learn the similarities between image regions. The self-representation graph learning model is derived from the self-similarity of image patches which is widely used in many image processing tasks such as de-noising, super-resolution. The local smoothness is also taken into consideration by using a Laplacian regularization; and (2) In [33], all the initial labels equally contribute to the classifying function but the global saliency confidence of each image region implied in the affinity graph is not taken into consideration. In our proposed method, we integrate the global saliency confidence prior implied in the learnt affinity matrix during the saliency ranking process, inducing a weighted manifold ranking saliency detection method.

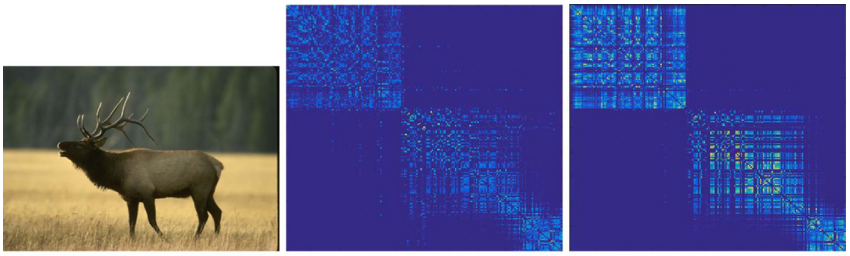
3. The proposed algorithm

As aforementioned, our proposed saliency detection algorithm consists of two parts. In the first part, we will learn an affinity matrix in an unsupervised manner. In the second part, we set image boundary regions as background queries and use traditional manifold ranking model described by Eq. (1) to obtain a coarse saliency map. Based on the coarse saliency map, we extract some regions which have been assigned higher saliency values as foreground queries. By taking the saliency confidence of each region obtained from the learned affinity matrix into consideration, we use a weighted manifold ranking model to estimate the final saliency map from the extracted foreground queries. In such a manner, the regions with higher saliency confidences will be assigned higher saliency values. The pipeline of the proposed approach is depicted in Fig. 1.

3.1. Affinity graph learning

Similar to previous graph based saliency methods, we first partition the input image I into n superpixels[3] $\mathcal{S} = \{s_1, \ldots, s_n\}$ by using the Simple Linear Iterative Clustering (SLIC) [57] algorithm.

[3] In this paper, each graph "node" represents an image "superpixel", so they refer to the same thing. We will use them interchangeably for different explanation purposes.



(a) (b) (c)

Fig. 2. Learning affinities of image region pairs. (a) Original image, (b) Traditional affinity matrix calculated by the Euclidean distance of feature vectors and Gaussian heat kernel function. (c) Affinity matrix learned by our unsupervised method.

For each superpixel s_i, we extract color and texture information to form a d-dimensional regional feature vector $\mathbf{x}_i \in \mathbb{R}^d$, then all the feature vectors of superpixels are ensembled to form a feature matrix $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \in \mathbb{R}^{d\times n}$. Based on self-representation theory, each feature vector x_i in $\mathbf{X}$ can be represented by a liner combination of others with different coefficients. The goal is to obtain a coefficient matrix $\mathbf{W} = \{\mathbf{w}_1, \ldots, \mathbf{w}_n\} \in \mathbb{R}^{n\times n}$, in which each $\mathbf{w}_i$ is the coefficients of $\mathbf{x}_i$. Traditional solution of obtaining $\mathbf{W}$ can be expressed as the following regularized problem:

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}} \|\mathbf{X} - \mathbf{X}\mathbf{W}\|_F^2 + \lambda\|\mathbf{W}\|_p, \tag{2}$$

where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix, and $\|\cdot\|_p$ represents p-norm of a matrix. In Eq. (2), the first term constraints the representation residual to be small, the second term is a regularization imposed on the coefficient matrix $\mathbf{W}$, e.g., sparsity, low-rank.

Generally, the coefficient vectors of similar original data should also be similar. Specifically, if $\mathbf{x}_i$ is very similar to $\mathbf{x}_j$, then the corresponding coefficient vectors $\mathbf{w}_i$ and $\mathbf{w}_j$ should be close to each other. This property can be constrained by the Laplacian smoothness criterion. For all data pairs, this criterion can be written as:

$$\sum_{i,j=1}^{n} \left\|\mathbf{w}_i - \mathbf{w}_j\right\|^2 z_{ij} = Tr\left(\mathbf{W}\mathbf{L}\mathbf{W}^T\right), \tag{3}$$

where $\mathbf{L}$ is the graph Laplacian of a given weight matrix $\mathbf{Z}$, i.e., $\mathbf{L} = \mathbf{D}^Z - \mathbf{Z}$, where $\mathbf{D}^Z$ is a diagonal matrix with $\mathbf{D}^Z_{ii} = \sum_{j=1}^{n} z_{ij}$.

By taking the Laplacian smoothness into consideration, we express our affinity graph learning model as follows

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}} \|\mathbf{X} - \mathbf{X}\mathbf{W}\|_F^2 + \lambda\|\mathbf{W}\|_F^2 + \beta Tr\left(\mathbf{W}\mathbf{L}^W\mathbf{W}^T\right), \tag{4}$$

where $\mathbf{L}^W$ is the Laplacian matrix of the affinity matrix $\mathbf{W}$, λ and β are two positive balance parameters. Here we enforce minimizing $\|\mathbf{W}\|_F^2$ for computation efficiency and preventing from the trivial solution. Other regularization such as $L_{2,1}$-norm for robust to outliers, L_1 norm for sparsity, and nuclear norm for low rank can be used to replace the Frobenius norm. If $\mathbf{L}^W$ is fixed, the minimization problem of Eq. (4) will be convex and have a closed form solution with a global minimum. However, $\mathbf{L}^W$ is unknown since the graph itself is unknown. To solve this problem, we propose an recursive method to get the optimal coefficient matrix (i.e., the affinity matrix). We start with an initial rough graph, and use its corresponding Laplacian matrix to impose smoothness of the coefficient vectors of the unknown matrix $\mathbf{W}$. Then the estimated $\mathbf{W}$ will be used to get a new Laplacian matrix for next iteration. In such a circular recursion, we can get the final optimal coefficient matrix $\mathbf{W}$.

Specifically, a rough weight matrix $\mathbf{Z}$ is computed in advance by using any traditional graph construction method, its Laplacian matrix is calculated by $\mathbf{L} = \mathbf{D}^Z - \mathbf{Z}$, then the coefficient matrix W is estimated by minimizing the following problem:

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}} \mathcal{H}(\mathbf{W}). \tag{5}$$

and

$$\begin{aligned}\mathcal{H}(\mathbf{W}) &= \|\mathbf{X} - \mathbf{X}\mathbf{W}\|_F^2 + \lambda\|\mathbf{W}\|_F^2 + \beta Tr\left(\mathbf{W}\mathbf{L}\mathbf{W}^T\right) \\ &= Tr\left((\mathbf{X} - \mathbf{X}\mathbf{W})^T(\mathbf{X} - \mathbf{X}\mathbf{W})\right) \\ &\quad + \lambda Tr\left(\mathbf{W}^T\mathbf{W}\right) + \beta Tr\left(\mathbf{W}\mathbf{L}\mathbf{W}^T\right).\end{aligned} \tag{6}$$

The solution of $\mathbf{W}$ can be obtained by differentiating $\mathcal{H}(\mathbf{W})$ with respect to $\mathbf{W}$ as follows;

$$\frac{\partial\mathcal{H}(\mathbf{W})}{\partial\mathbf{W}} = -2\mathbf{X}^T\mathbf{X} + 2\mathbf{X}^T\mathbf{X}\mathbf{W} + 2\lambda\mathbf{W} + 2\rho\mathbf{W}\mathbf{L}. \tag{7}$$

We set Eq. (7) to 0, and obtain

$$\left(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}\right)\mathbf{W} + \mathbf{W}(2\rho\mathbf{L}) = \mathbf{X}^T\mathbf{X}, \tag{8}$$

where $\mathbf{I}$ is a $n\times n$ identity matrix. Eq. (8) is a Sylvester equation, and we solve it using the *sylvester* function of Matlab to get $\mathbf{W}$.

Finally, the affinity matrix $\mathbf{A}$ is obtained by $\mathbf{A} = \frac{|\mathbf{W}| + |\mathbf{W}^T|}{2}$.

The whole procedure to obtain $\mathbf{A}$ can be summarized in Algorithm. 1. Note that the learned affinity matrix $\mathbf{A}$ can be regarded as a full connected graph, we need to remove some redundant connection of graph nodes. Similar to previous methods [32,34], each node is connected to those nodes neighboring it and the nodes sharing common boundaries with its neighboring nodes. All the nodes on the four sides of image are also connected and each node is not connected with itself.

Algorithm 1 Unsupervised affinity graph learning.

Input: Original feature matrix $\mathbf{X}$ and an initial affinity matrix $\mathbf{A}_0$
Initialization: Parameters λ and β, $t = 0$, $\varepsilon = 0.00001$
Repeat
1. Calculate the graph Laplacian of $\mathbf{A}_0$, $\mathbf{L}_t$=$\mathbf{D}_t - \mathbf{A}_t$;
2. Get $\mathbf{W}_{t+1}$ by solving Eq. (8);
3. Update $\mathbf{A}$ by $\mathbf{A}_{t+1} = \frac{1}{2}\left(|\mathbf{W}_{t+1}| + \left|\mathbf{W}_{t+1}{}^T\right|\right)$;
4. $t = t + 1$;
Until $\|\mathbf{W}_t - \mathbf{W}_{t+1}\|_F^2 < \varepsilon$
Output: Affinity matrix $\mathbf{A}$

Fig. 2 shows an example of our learned affinity matrix. Fig. 2(a) is an input image, it is composed of three main parts: the dark green forest, the sika deer and the yellow meadow. Fig. 2(b) is the traditional affinity matrix calculated by the Euclidean distance of feature vectors and Gaussian heat kernel function, as can be seen, it contains some noisy values and the relevance of regions from the same image part are not very strong (i.e., the affinity values between similar image regions are not large enough). On the other hand, our method can learn a clearer block-diagonal affinity matrix, noisy values can be efficiently suppressed and the relevance

(a) (b) (c)

Fig. 3. An example of saliency confidences implied in the affinity matrix. (a) Original image, (b) degree value of each superpixel, (c) final saliency map obtained by using the learned affinity matrix.

Fig. 4. Quantitative comparisons of saliency maps generated by different methods.

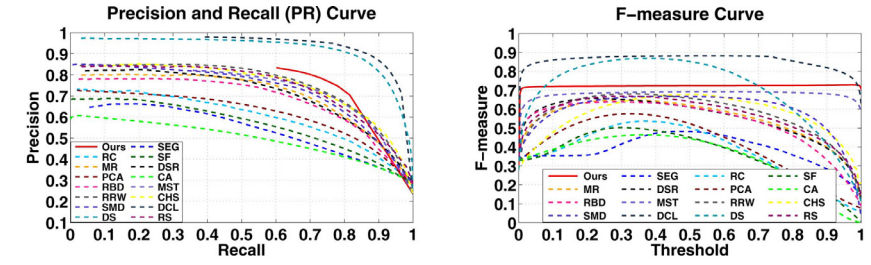
of similar image regions are stronger, as shown in Fig. 2(c). This property can benefit the following saliency ranking and make the salient parts more separable from background.

3.2. Ranking from background seeds

It is observed that background often presents local or global appearance connectivity with the four image boundaries. This boundary prior observation has been used in many previous saliency detection works [32,34,35] and verified more general and effective than previously used center prior [58,59]. In this work, we also follow this boundary prior and assume that the four image boundaries as background. Note that sometimes the salient objects may also touch image boundaries, and some works try to reduce the salient foreground noises in the image boundary regions object [34,35,60] and improve the final saliency detection results. But their is no algorithm which can absolutely remove the foreground noises from image boundaries because distinguishing salient regions and background regions is a difficult problem in itself. How to remove salient noisy areas from boundaries is not the focus of this work. In this step, similar to the work in [32], we use the nodes on each side of image as labelled background queries, then compute the saliency of other nodes based on their relevances to those queries as background labels by traditional manifold ranking. The four ranked maps from the background queries are then integrated to generate a coarse saliency map, as shown in Fig. 1. For this first ranking step, we only aim to obtain a coarse saliency map by setting image boundary regions as background queries, i.e., the boundary regions are set as background queries with probability 1 (no weight needed). For simplicity, we use traditional manifold ranking for initial saliency estimation. In fact, we have tested the initial saliency estimation by using weighted manifold ranking, but it makes little sense to the final results.

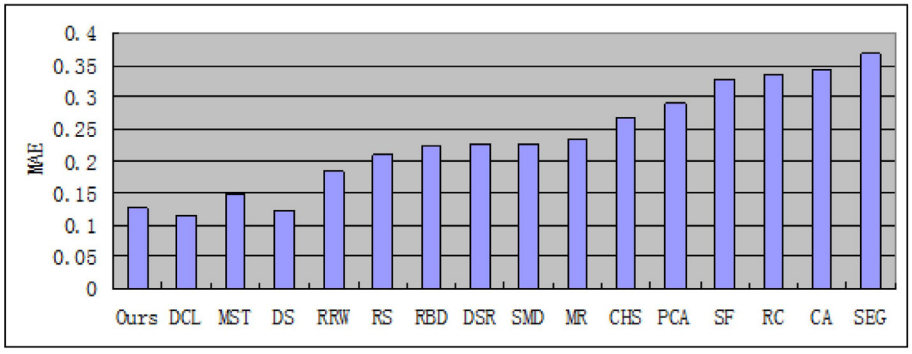
3.3. Weighted ranking from foreground seeds

In the coarse saliency map obtained from the second step, some foreground areas are not fully highlighted and some background areas are falsely detected as foreground. Previous method [32] directly applies binary segmentation on the coarse saliency map from the first stage, and take the labelled foreground nodes as salient queries. The saliency of each node is computed based on its relevance to foreground queries for the final map. In this manner, the saliency confidences of image regions implied in the affinity matrix are not fully exploited. In our work, in order to fully capture the salient regions, we simply set the first half of superpixels with higher saliency values in the coarse saliency map as salient foreground seeds, and the final saliency of each region is obtained by ranking with weight which is modelled by saliency confidences of foreground seeds. We model the saliency confidence of each region i as its degree value d_{ii} on the affinity graph. Given the learned affinity matrix $\mathbf{A}$, in which $\mathbf{A}_{ij}$ represents the similarity between region i and j. Thus the ith row/column represents the similarity between image region i and other regions. From the perspective of global contrast, if a region belongs to a salient object, it should have high contrast (low similarity) with most of other regions, i.e., $\mathbf{A}_{ij}(j = 1 \ldots n, j \neq i)$ should be small for most j. Therefore, for the degree matrix of $\mathbf{A}$, d_{ii} should be small if image region i belongs to a salient object. This property

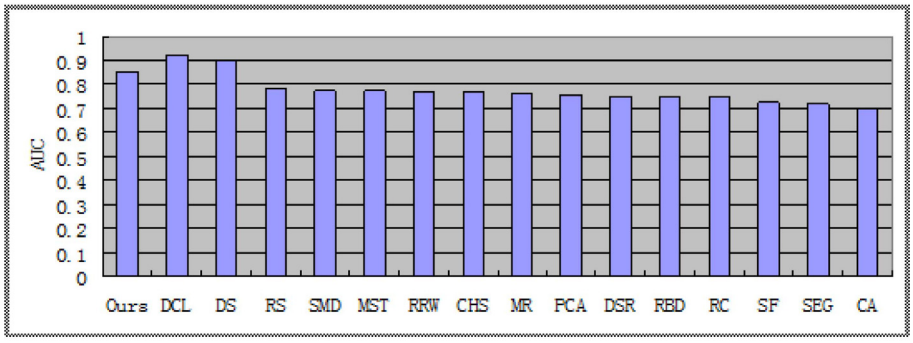


(a)

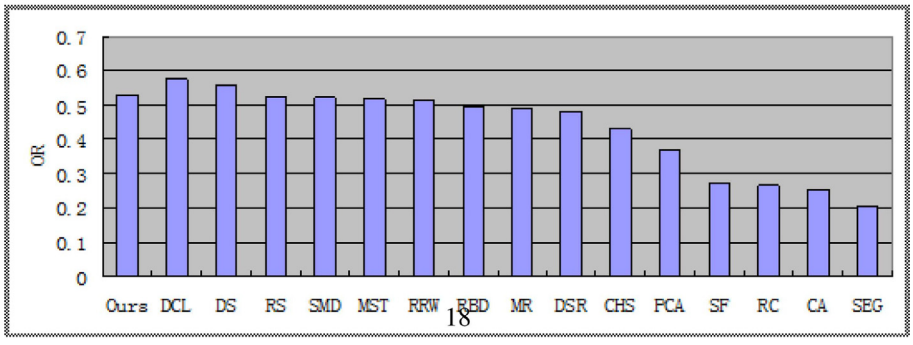
(b)



(c)

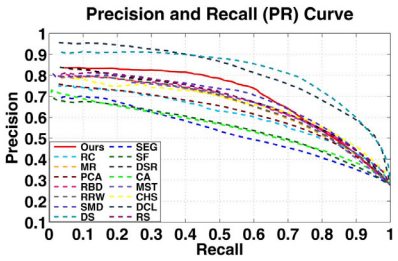


(d)

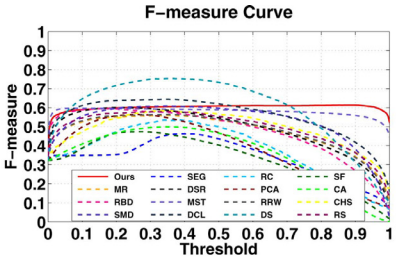


(e)

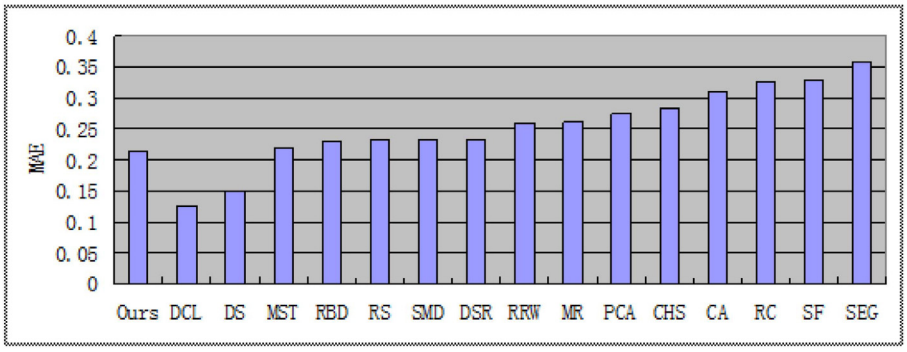
Fig. 5. Quantitative performance comparison of different methods on the ECSSD dataset. (a) Precision-recall curves, (b) F-measure curves, (c) MAE values, (d) AUC values, and (e) OR values.



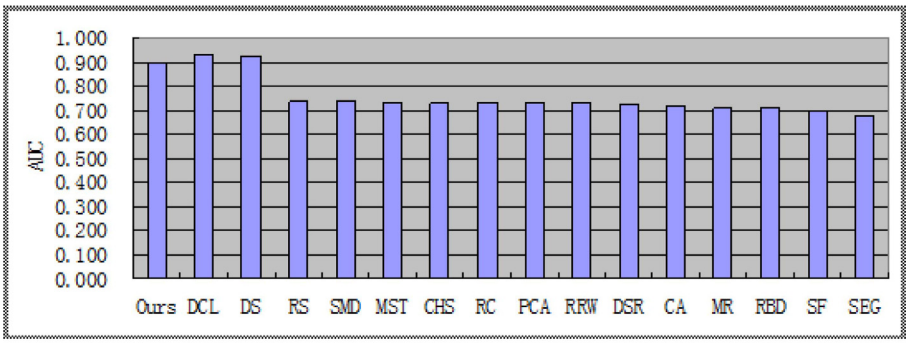
(a)



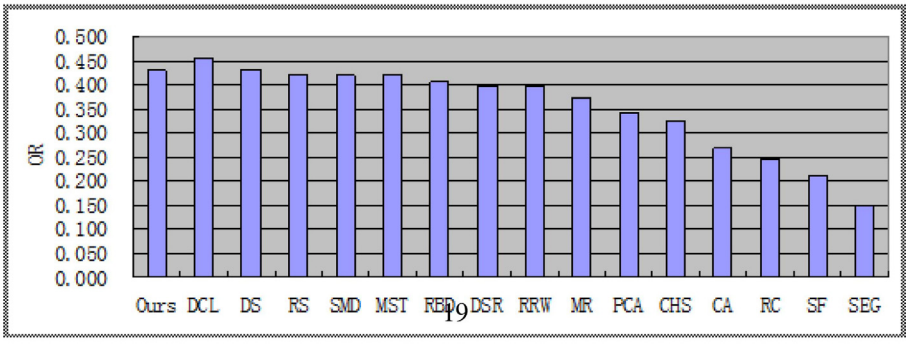
(b)



(c)



(d)

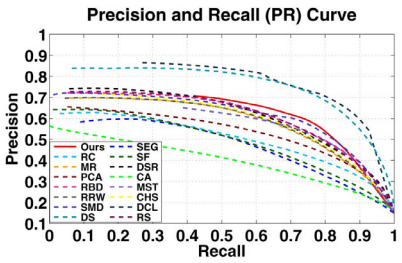


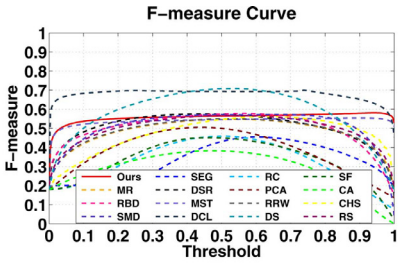
(e)

Fig. 6. Quantitative performance comparison of different methods on the SOD dataset. (a) Precision-recall curves, (b) F-measure curves, (c) MAE values, (d) AUC values, and (e) OR values.

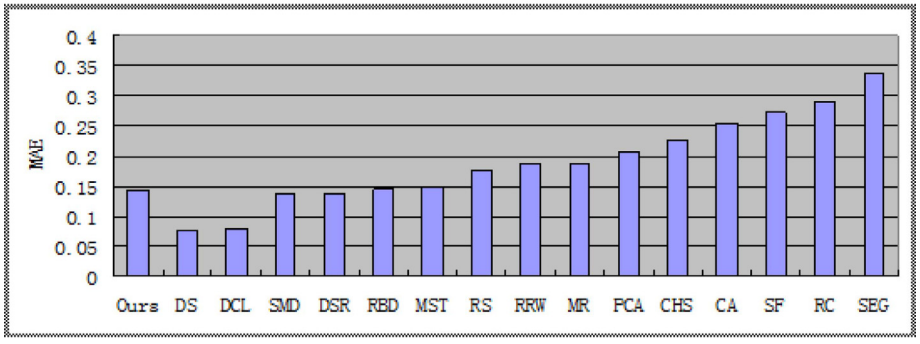
implied in the affinity matrix can be used as the saliency confidence of different image regions. Fig. 3 gives an example, Fig. 3(a) is an input image, the degree value of each superpixel is shown in Fig. 3(b). As can be seen, salient part (the rabbit) of the image will have smaller degree values, which verifies that the global contrast information is implied in the affinity matrix and it can be used to improve traditional saliency ranking results.

By taking the saliency confidences into consideration, the superpixels with higher saliency confidences (with smaller degree values) will be assigned higher saliency values in the final saliency

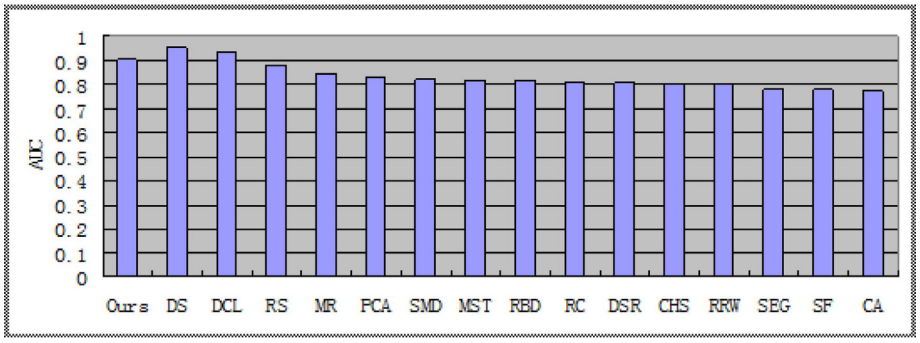




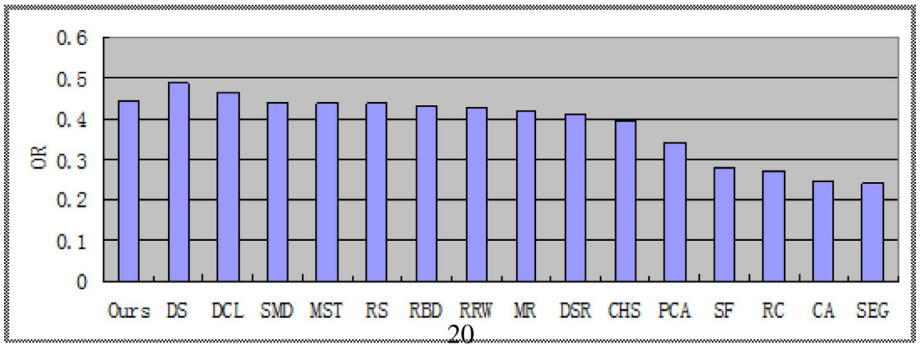
(a) (b)



(c)



(d)



(e)

Fig. 7. Quantitative performance comparison of different methods on the DUT-OMRON dataset. (a) Precision-recall curves, (b) F-measure curves, (c) MAE values, (d) AUC values, and (e) OR values.

map. In addition, in order to reduce ranking errors, we also set the other part of superpixels with smaller saliency values in the coarse saliency map as sink points during our weighted ranking process. Specifically, after we obtain the coarse saliency map, we compute a threshold value T by Otsu [61], the regions with saliency values smaller than T are set as sink points. Finally, we derive the weighted MR model as follows:

$$\hat{f} = \arg\min_{f} \frac{1}{2} \times \left(\sum_{i,j=1}^{n} w_{ij} \left\| \frac{\delta_i f_i}{\sqrt{d_{ii}}} - \frac{\delta_j f_j}{\sqrt{d_{jj}}} \right\|^2 + \mu \sum_{i=1}^{n} \left\| \delta_i f_i - d_{ii}^{m} y_i \right\|^2 \right), \quad (9)$$

where m determines the degree of saliency confidences on initial labels, and it is set to 1/2 in our work, enabling a salient superpixel to have a relatively higher confidence than a background superpixel. $\delta_i = 0$ if superpixel i is a sink point and 1 otherwise. The sink points be intuitively imagined as "black holes", where ranking scores spreading to them will be absorbed and no ranking scores would escape from them [60,62]. This can be understood in this way: the superpixels which are set as sink points are those with smaller saliency values in the coarse map, they are very likely to be background. By setting appropriate sink points, these background regions can be effectively suppressed during the ranking process.

By writing Eq. (9) in its matrix form and differentiating the function respect to $\mathbf{f}$, we have

$$\left(\mathbf{I} - \alpha \mathbf{D}^{-1}\mathbf{A}\mathbf{I}_s\right)\mathbf{f} = (1-\alpha)\mathbf{D}^{-\frac{1}{2}}\mathbf{y}, \quad (10)$$

where $\alpha = 1/(1+\mu)$ and $\mathbf{A}$ is the affinity matrix learned in Section 3.1. $\mathbf{I}_s = diag(\delta_1, \delta_2, \ldots \delta_n)$ is the sink point indicator matrix.

From Eq. (10), the relevance scores between labeled and unlabeled graph vertices can be obtained as

$$\mathbf{f} \propto \left(\mathbf{I} - \alpha \mathbf{D}^{-1}\mathbf{A}\mathbf{I}_s\right)^{-1}\mathbf{D}^{-\frac{1}{2}}\mathbf{y}. \quad (11)$$

Finally, the saliency value of each superpixel $S(i)$ can be directly assigned by the ranking score:

$$S(i) = f(i), i = 1, 2, \ldots, n. \quad (12)$$

Similar to [63], we also apply an adaptive contrast enhancement for the final saliency map with the following sigmoid mapping:

$$g(x) = \frac{1}{1 + \exp\left(-\gamma(x - \tau)\right)}, \quad (13)$$

where τ is an adaptive threshold obtained using Otsu's binary threshold method [61] and γ controls the overall sharpness. We also set $\gamma = 20$ in our experiments.

4. Experimental results

4.1. Experiment setup

For most of the saliency detection datasets, the image size is around 300×400 pixels. In order to balance the computation efficiency and accuracy, the number of segmented super-pixels is often set to 200 as to the image size in most of previous works. Empirical experiments also demonstrate that the final results change little while the number of segmented super-pixels varies between 150 and 300 when the image size is about 300×400 pixels. Therefore, in order to keep consistent with previous works, we also set the number of segmented super-pixels to 200 in our work, i.e., $n = 200$ in all the experiments. In the affinity graph learning step, λ and β are all set to 1 empirically, the effects of λ and β on the learned affinity graph matrix is out of the scope of this work. $\alpha = 1/(1+\mu)$ is set to 0.99 as previous works [32,35]. In our experiment, both color and texture features are used, including RGB color, Lab color, HSV color, opponent color [64] and local binary pattern (LBP) [65]. For each superpixel, we use the mean value of each feature to assemble a 13-dimensional feature vector. All the feature vectors of superpixels are then used for affinity graph learning.

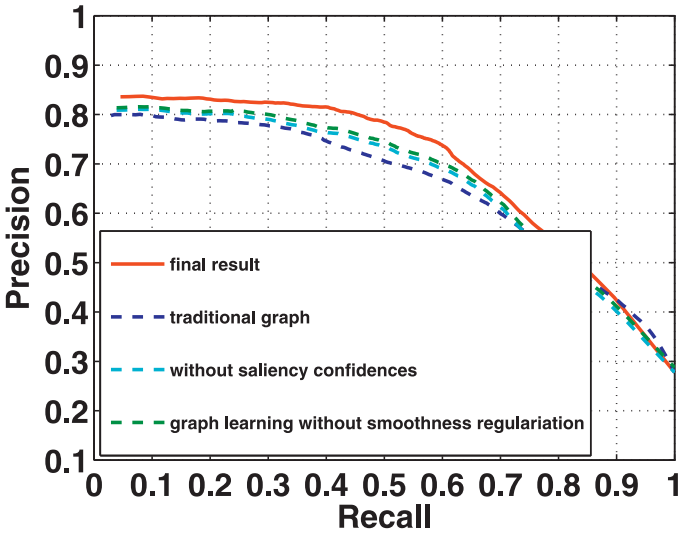


Fig. 8. PR curve comparison of each individual component on SOD dataset.

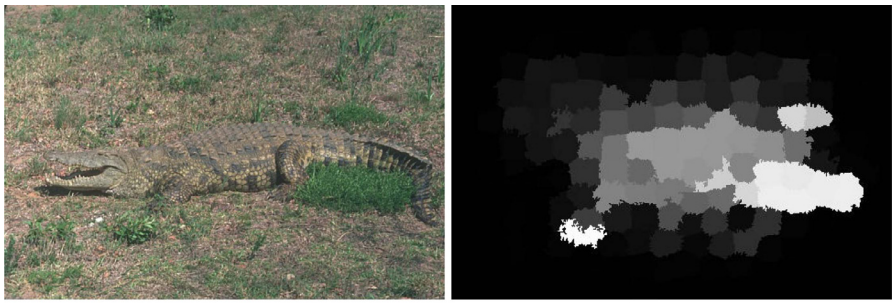
4.2. Datasets

The performance evaluation is conducted on three challenging datasets: SOD [66], ECSSD [67] and DUT-OMRON [32]. SOD consists of 300 images collected from the Berkeley segmentation dataset and most of which have multiple salient objects. ECSSD contains 1000 semantically meaningful but structurally complex images. DUT-OMRON is another challenging dataset which contains 5168 images with complex background.

4.3. Comparison with state-of-the-art methods

In the experiments, we qualitatively and quantitatively compare the proposed approach with fifteen state-of-the-art approaches, including thirteen unsupervised methods: SEG [68], RC [69], SF [70], MR [32], DSR [71], PCA [72], CA [73], RBD [74], RRW [34], CHS [17], MST [63], SMD [75] and RS [33], and two supervised deep learning based methods: DS [20] and DCL [22]. Fig. 4 shows the qualitative comparisons of saliency maps generated by our approach and other different methods. As can be seen, our method can clearly highlight the salient objects in an image, the background areas are efficiently suppressed. This verifies that the weighted manifold ranking can assign higher saliency values to those superpixles which are with higher saliency confidences implied in the affinity matrix, or vice versa. Due to the fact that the learned affinity matrix can reveal the intrinsic relevance between different superpixels, our approach is also applicable for images with complex background.

For quantitative evaluation, we use precision-recall curve, F-measure curve, mean absolute error (MAE), area under curve (AUC) and overlapping ratio (OR) to evaluate the performance of our algorithm. The precision and recall scores are computed by binarizing the estimated saliency maps with a threshold sliding from 0 to 255 and compare the binary maps with ground truth maps.



(a) (b)

Fig. 9. Failure case of our method. (a) Original image, (b) saliency detection result of our method.

Usually, precision and recall are both important and therefore F-measure is used as the overall performance measure which can be written as:

$$F_\beta = \frac{\left(1+\beta^2\right)\cdot precision \cdot recall}{\beta^2 \cdot precision + recall}, \tag{14}$$

where β^2 is set to 0.3 as suggested in [76] to emphasize precision. As neither precision nor recall measure evaluate the true negative saliency assignments, we also use the mean absolute error (MAE) as a complementary. The MAE score calculates the average difference between the saliency map M and the ground truth GT, it is computed as:

$$MAE = \frac{1}{H \times W}\sum_{i=1}^{H\times W} |M(i) - GT(i)|, \tag{15}$$

where H and W are the height and width of the input image, respectively. Meanwhile, we use true positive and false positive rates to calculate the AUC score. OR is defined as the overlapping ratio between the segmented object mask S' and ground truth G : $OR = |S' \cap G|/|S \cup G|$, where S' is obtained by binarizing S using an adaptive threshold, i.e., twice the mean values of S as in [77].

Quantitative evaluation results of different methods on three datasets are shown in Figs. 5–7, respectively. As can be seen, our approach can outperform all other state-of-the-art unsupervised methods. Note that for both of the three datasets, our method can perform better than all of the previous graph based methods such as MR, RRW and RS, which demonstrate the validity of the proposed affinity graph learning and weighted manifold ranking. Compared to deep learning based methods, our approach can not win the performance. However, deep learning based methods need large amount of training examples, while our method is totally unsupervised.

In order to verify the superiority of our learned affinity matrix and the effect of the global saliency confidences, we also give the result which is obtained by using traditional Euclidean distance and Gaussian kernel function based affinity matrix and the result without global saliency confidences. We show the precision-recall curves of SOD dataset in Fig. 8. As can be seen, the learned affinity matrix can significantly promote the final results. The global confidences implied in the affinity matrix can also benefit the final results. In addition, in order to verify the effect of the Laplacian regularization term used in the affinity graph learning model, we also plot the precision-recall curve result by using the affinity graph learned without the Laplacian regularization. As can be seen, the Laplacian regularization term can also facilitate the final results.

4.4. Running time

For testing the running time of our method, we implemented it by using Matlab R2016a on a laptop with Intel Core i5-4200M 2.5 GHz CPU and 8 GB RAM. The average running time of an image with size 400×300 is about 1.4 s on ECSSD dataset. The mainly time cost parts are the affinity matrix learning process and superpixel segmentation, it spends about 0.4 s to generate the superpixels and 0.7 s to learn the affinity matrix, the actual saliency computation spends only about 0.3 s.

4.5. Limitation and analysis

Fig. 9 shows a failure case, where the background and foreground in original image is very similar. In such a case, there is small difference between background and foreground, and the similarity graph of image regions is hard to learn. However, it is also challenging for the state-of-the-arts even deep learning based methods. Objectness measure may be used to solve this problem. However, the current objectness measure methods are not accurate enough. In existing methods, different features are often extracted for saliency detection, but they concatenated different features directly. Since different features focus different properties of an image, the feature spaces and scales are different, directly concatenating may not leverage the complementary information of different features. Thus, in our future work, we intend to learn the similarity graph in a multi-view perspective.

5. Conclusion and future work

We present a bottom-up approach to detect salient regions in images through weighted manifold ranking on a learned graph. Different from traditional graph based saliency detection methods, we use an unsupervised manner to learn the affinity matrix of the graph which is composed by image superpixels. Our learned affinity matrix can better reveal the intrinsic relevances between superpixels. By considering the global saliency confidences implied in the learned affinity matrix, we propose a saliency confidences weighted manifold ranking for final saliency detection, this makes those superpixels with higher saliency confidences will be assigned higher saliency values in the final saliency map and background elements can be efficiently suppressed.

We have also realized that deep learning has obtained great success in the area of saliency detection. Thus we are going to combine deep learning for saliency detection in our future work.

Acknowledgments

This work is partly supported by the Fundamental Research Funds for the Central Universities, China University of Geosciences

(Wuhan) under Grant No. CUG170654 and the National Natural Science Foundation of China under Grant No. (61701451 and 61601261). In our experiments, for the sake of efficiency, we also used GPU for computation and would like to thank NVIDIA Corporation for the donation of a Titan Xp GPU card used in this research.

References

[1] B. Alexe, T. Deselaers, V. Ferrari, Measuring the objectness of image windows, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2189–2202.
[2] L. Itti, C. Koch, A saliency-based search mechanism for overt and covert shifts of visual attention, Vision Res. 40 (1012) (2000) 1489–1506.
[3] U. Rutishauser, D. Walther, C. Koch, P. Perona, Is bottom-up attention useful for object recognition? in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, IEEE, 2004, pp. 37–44.
[4] G. Sharma, F. Jurie, C. Schmid, Discriminative spatial saliency for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 3506–3513.
[5] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, S.-M. Hu, Sketch2Photo: internet image montage, ACM Trans. Gr. 28 (5) (2009) 124.
[6] P. Wang, J. Wang, G. Zeng, J. Feng, H. Zha, S. Li, Salient object detection for searched web images via global saliency, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 3194–3201.
[7] P. Hiremath, J. Pujari, Content based image retrieval using color boosted salient points and shape features of an image, Int. J. Image Process. 2 (1) (2008) 10–17.
[8] K.-Y. Chang, T.-L. Liu, S.-H. Lai, From co-saliency to co-segmentation: an efficient and fully unsupervised energy minimization model, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 2129–2136.
[9] C. Rother, V. Kolmogorov, A. Blake, Grabcut: interactive foreground extraction using iterated graph cuts, ACM Trans. Gr. 23 (3) (2004) 309–314.
[10] S. Goferman, L. Zelnik-Manor, A. Tal, Context-aware saliency detection, IEEE Trans. Pattern Anal. Mach. Intell. 34 (10) (2012) 1915–1926.
[11] L. Marchesotti, C. Cifarelli, G. Csurka, A framework for visual saliency detection with applications to image thumbnailing, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE, 2009, pp. 2232–2239.
[12] R. Margolin, L. Zelnik-Manor, A. Tal, Saliency for image manipulation, Visual Comput. 29 (5) (2013) 381–392.
[13] J. Sun, H. Ling, Scale and object aware image thumbnailing, Int. J. Comput. Vision 104 (2) (2013) 135–153.
[14] B. Alexe, T. Deselaers, V. Ferrari, What is an object? in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2010, pp. 73–80.
[15] J. Yang, M.-H. Yang, Top-down visual saliency via joint CRF and dictionary learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 2296–2303.
[16] N. Tong, H. Lu, Y. Zhang, X. Ruan, Salient object detection via global and local cues, Pattern Recognit. 48 (10) (2015) 3258–3267.
[17] Q. Yan, L. Xu, J. Shi, J. Jia, Hierarchical saliency detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1155–1162.
[18] C. Tang, C. Hou, P. Wang, Z. Song, Salient object detection using color spatial distribution and minimum spanning tree weight, Multimed. Tools Appl. 75 (12) (2016) 6963–6978.
[19] C. Tang, J. Wu, C. Zhang, P. Wang, W. Li, Salient object detection via weighted low rank matrix recovery, IEEE Signal Process. Lett. 24 (4) (2017) 490–494.
[20] X. Li, L. Zhao, L. Wei, M.H. Yang, F. Wu, Y. Zhuang, H. Ling, J. Wang, Deepsaliency: multi-task deep neural network model for salient object detection, IEEE Trans. Image Process. 25 (8) (2016) 3919.
[21] L. Wang, L. Wang, H. Lu, P. Zhang, R. Xiang, Saliency detection with recurrent fully convolutional networks, in: Proceedings of the European Conference on Computer Vision, 2016, pp. 825–841.
[22] G. Li, Y. Yu, Deep contrast learning for salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 478–487.
[23] H. Li, J. Chen, H. Lu, Z. Chi, CNN for saliency detection with low-level feature integration, Neurocomputing 226 (2017) 212–220.
[24] W. Lijun, L. Huchuan, W. Yifan, F. Mengyang, W. Dong, Y. Baocai, R. Xiang, Learning to detect salient objects with image-level supervision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.
[25] L. Zhiming, M. Akshaya, A. Andrew, E. Justin, L. Shaozi, J. Pierre-Marc, Non-local deep features for salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.
[26] A. Borji, M.-M. Cheng, H. Jiang, J. Li, Salient object detection: a survey, arXiv:1411.5878, 2014.
[27] M. Cheng, N.J. Mitra, X. Huang, P.H. Torr, S. Hu, Global contrast based salient region detection, IEEE Trans. Pattern Anal. Mach. Intell. 37 (3) (2015) 569–582.
[28] A. Borji, M.-M. Cheng, H. Jiang, J. Li, Salient object detection: a benchmark, IEEE Trans. Image Process. 24 (12) (2015) 5706–5722.
[29] J. Harel, K. Christof, P. Perona, Graph-based visual saliency, in: Proceedings of the Neural Information Processing Systems, 2006, pp. 545–552.
[30] V. Gopalakrishnan, Y. Hu, D. Rajan, Random walks on graphs for salient object detection in images, IEEE Trans. Image Process. 19 (12) (2010) 3232–3242.
[31] K.Y. Chang, T.L. Liu, H.T. Chen, S.H. Lai, Fusing generic objectness and visual saliency for salient object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2011, pp. 914–921.
[32] C. Yang, L. Zhang, H. Lu, X. Ruan, Saliency detection via graph-based manifold ranking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 3166–3173.
[33] L. Zhang, C. Yang, H. Lu, R. Xiang, M.H. Yang, Ranking saliency, IEEE Trans. Pattern Anal. Mach. Intell. 39 (9) (2017) 1892–1904.
[34] C. Li, Y. Yuan, W. Cai, Y. Xia, Robust saliency detection via regularized random walks ranking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2710–2717.
[35] Q. Wang, W. Zheng, R. Piramuthu, Grab: visual saliency via novel graph model and background priors, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 535–543.
[36] C. Tang, J. Wu, Y. Hou, P. Wang, W. Li, A spectral and spatial approach of coarse-to-fine blurred image region detection, IEEE Signal Process. Lett. 23 (11) (2016) 1652–1656.
[37] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Trans. Pattern Anal. Mach. Intell. 23 (11) (2001) 1222–1239.
[38] X. Wang, Y. Tang, S. Masnou, L. Chen, A global/local affinity graph for image segmentation, IEEE Trans. Image Process. 24 (4) (2015). 1399–411
[39] F. Dornaika, T.Y. El, Learning flexible graph-based semi-supervised embedding, IEEE Trans. Cybern. 46 (1) (2016) 206–218.
[40] R. Luo, W. Liao, X. Huang, Y. Pi, Feature extraction of hyperspectral images with semisupervised graph learning, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. (2016) 1–11.
[41] A.Y. Ng, M.I. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: Proceedings of the Neural Information Processing Systems, 2002, pp. 849–856.
[42] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comput. 15 (15) (2003) 1373–1396.
[43] F. Nie, X. Wang, H. Huang, Clustering and projected clustering with adaptive neighbors, in: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2014, pp. 977–986.
[44] E. Elhamifar, R. Vidal, Sparse subspace clustering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 2790–2797.
[45] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma, Robust recovery of subspace structures by low-rank representation, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 171–184.
[46] D.L. Donoho, Compressed sensing, IEEE Trans. Inf. Theory 52 (4) (2006) 1289–1306.
[47] L. Qiao, S. Chen, X. Tan, Sparsity preserving projections with applications to face recognition, Pattern Recognit. 43 (1) (2010) 331–341.
[48] B. Cheng, J. Yang, S. Yan, Y. Fu, T.S. Huang, Learning with *ellsp*1-graph for image analysis, IEEE Trans. Image Process. 19 (4) (2010) 858–866.
[49] C. Wang, S. Yan, L. Zhang, H.J. Zhang, Multi-label sparse coding for automatic image annotation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1643–1650.
[50] X. Zhang, Non-negative low rank and sparse graph for semi-supervised learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2328–2335.
[51] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 3360–3367.
[52] J. Waqas, Z. Yi, L. Zhang, Collaborative neighbor representation based classification using l 2-minimization approach, Pattern Recognit. Lett. 34 (2) (2013) 201208.
[53] F. Dornaika, A. Bosaghzadeh, H. Salmane, Y. Ruichek, Graph-based semi-supervised learning with local binary patterns for holistic object categorization, Expert Syst. Appl. 41 (17) (2014) 7744–7753.
[54] Z. Dengyong, W. Jason, G. Arthur, B. Olivier, S. Bernhard, Ranking on data manifolds, in: Proceedings of the Neural Information Processing Systems, 2003, pp. 169–176.
[55] B.S. Brin, L. Page, The anatomy of a large scale hypertextual web search engine, Comput. Netw. ISDN Syst. 30 (1–7) (1998) 107–117.
[56] K. Fu, I.Y.H. Gu, J. Yang, Learning full-range affinity for diffusion-based saliency detection, in: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 2016, pp. 1926–1930.
[57] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Susstrunk, Slic superpixels compared to state-of-the-art superpixel methods, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2274–2282.
[58] T. Judd, K. Ehinger, F. Durand, A. Torralba, Learning to predict where humans look, in: Proceedings of the IEEE International Conference on Computer Vision, 2009, pp. 2106–2113.
[59] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, H.Y. Shum, Learning to detect a salient object, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2) (2011) 353–367.
[60] S. Chen, L. Zheng, X. Hu, P. Zhou, Discriminative saliency propagation with sink points, Pattern Recognit. 60 (2016) 2–12.
[61] N. Ohtsu, A threshold selection method from gray-level histograms, Automatica 11 (285–296) (1975) 23–27.
[62] X.Q. Cheng, P. Du, J. Guo, X. Zhu, Y. Chen, Ranking on data manifold with sink points, IEEE Trans. Knowl. Data Eng. 25 (1) (2013) 177–191.

[63] W.-C. Tu, S. He, Q. Yang, S.-Y. Chien, Real-time salient object detection with a minimum spanning tree, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2334–2342.
[64] S. Frintrop, T. Werner, G.M. Garcıa, Traditional saliency reloaded: a good old model in new shape, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 82–90.
[65] T. Ojala, I. Harwood, A comparative study of texture measures with classification based on feature distributions, Pattern Recognit. 29 (1) (1996) 51–59.
[66] V. Movahedi, J.H. Elder, Design and perceptual validation of performance measures for salient object segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2010, pp. 49–56.
[67] J. Shi, Q. Yan, L. Xu, J. Jia, Hierarchical image saliency detection on extended CSSD, IEEE Trans. Pattern Anal. Mach. Intell. 38 (4) (2015) 717–729.
[68] E. Rahtu, J. Kannala, M. Salo, J. Heikkil, Segmenting Salient Objects from Images and Videos, Springer, Berlin Heidelberg, 2010.
[69] M. Cheng, N.J. Mitra, X. Huang, P.H.S. Torr, S. Hu, Global contrast based salient region detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 409–416.
[70] F. Perazzi, P. Krahenbuhl, Y. Pritch, A. Hornung, Saliency filters: Contrast based filtering for salient region detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 733–740.
[71] X. Li, H. Lu, L. Zhang, R. Xiang, M.H. Yang, Saliency detection via dense and sparse reconstruction, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 2976–2983.
[72] M. Ran, A. Tal, L. Zelnikmanor, What makes a patch distinct? in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1139–1146.
[73] S. Goferman, L. Zelnikmanor, A. Tal, Context-aware saliency detection, IEEE Trans. Pattern Anal. Mach. Intell. 34 (10) (2010) 1915–1926.
[74] W. Zhu, S. Liang, Y. Wei, J. Sun, Saliency optimization from robust background detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2814–2821.
[75] H. Peng, L. Bing, H. Ling, W. Hu, Salient object detection via structured matrix decomposition, IEEE Trans. Pattern Anal. Mach. Intell. 39 (4) (2017) 818–832.
[76] R. Achanta, S. Hemami, F. Estrada, S. Susstrunk, Frequency-tuned salient region detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1597–1604.
[77] X. Li, Y. Li, C. Shen, A. Dick, A.V.D. Hengel, Contextual hypergraph modeling for salient object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2014, pp. 3328–3335.

Xinzhong Zhu is a professor at College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, PR China. He received his B.S. degree in computer science and technology from the University of Science and Technology Beijing (USTB), Beijing, in 1998, and received M.S. degree in software engineering from National University of Defense Technology in 2005, and now he is a Ph.D. candidate at XIDIAN University. From August 2013, he spent one and a half years in visiting the Multimedia Software Engineering Research Centre (MERC) at City University of Hong Kong. His research interests include machine learning, computer vision, manufacturing informatization, robotics and system integration, and intelligent manufacturing. He is a member of the ACM.

Chang Tang received his Ph.D. degree from Tianjin University, Tianjin, China in 2016. He joined the AMRL Lab of the University of Wollongong between Sep. 2014 and Sep. 2015. He is now an associate professor at the School of Computer Science, China University of Geosciences, Wuhan. His current research interests include machine learning and data mining.

Pichao Wang received the B.E. degree in network engineering from Nanchang University, Nanchang, China, in 2010, and the M.S. degree in communication and information system from Tianjin University, Tianjin, China, in 2013. He is currently working toward the Ph.D. degree with the School of Computing and Information Technology, University of Wollongong, Wollongong, Australia. His current research interests include computer vision and machine learning.

Huiying Xu is an associate professor at College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, PR China. She received his B.S. degree in computer science and technology from the Zhejiang Normal University, in 1999, and received M.S. degree in software engineering from National University of Defense Technology in 2005. Her research interests include kernel learning and feature selection, machine learning, computer vision, image processing, pattern recognition, computer simulation, digital watermarking, and their applications. She is a member of the China Computer Federation.

Minhui Wang is now a pharmacist at the Department of Pharmacy, People's Hospital of Lian'shui County. Her current research interests include medical data analysis and effects of drugs on vision.

Jie Tian received the PhD degree from the Institute of Automation, Chinese Academy of Sciences, China, in 1992. He is a fellow of IEEE and works as a professor at School of Electronic Engineering, XIDIAN University. His research interests include the medical image process and analysis and pattern recognition. Dr. Tian is the Beijing Chapter Chair of The Engineering in Medicine and Biology Society of the IEEE.