Signal Processing: Image Communication 47 (2016) 380–401

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

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

Multi-scale salient object detection using graph ranking and global–local saliency refinement

Idir Filali [a], Mohand Saïd Allili [b,*], Nadjia Benblidia [a]

[a] *LRDSI Laboratory, Saad Dahlab University – Blida1, Blida, Algeria*
[b] *Department of Computer Science and Engineering, University of Quebec in Outaouais, Gatineau, QC, Canada J8X 3X7*

ARTICLE INFO

Article history:
Received 15 January 2016
Received in revised form
28 July 2016
Accepted 28 July 2016
Available online 5 August 2016

Keywords:
Salient object detection (SOD)
Multi-layer graphs
Random forests
Region and boundary information
Feature relevance

ABSTRACT

We propose an algorithm for salient object detection (SOD) based on multi-scale graph ranking and iterative local–global object refinement. Starting from a set of multi-scale image decompositions using superpixels, we propose an objective function which is optimized on a multi-layer graph structure to diffuse saliency from image borders to salient objects. This step aims at roughly estimating the location and extent of salient objects in the image. We then enhance the object saliency through an iterative process employing random forests and local boundary refinement using color, texture and edge information. We also use a feature weighting scheme to ensure optimal object/background discrimination. Our algorithm yields very accurate saliency maps for SOD while maintaining a reasonable computational time. Experiments on several standard datasets have shown that our approach outperforms several recent methods dealing with SOD.

1. Introduction

Salient object detection is an important problem for several computer vision applications such as object detection and segmentation [2,18,41], content-based image retrieval [10,24], image editing [16], pose estimation [48], image/video summarization classification [51,62] and GPS location estimation [45]. Humans are capable of quickly and accurately identifying salient objects in images where usually the attention is drawn. Saliency in general expresses rarity and local contrast of image attributes such as color, edges and texture [10]. Early saliency models have been developed mainly for eye fixation prediction in natural images aiming to understand human visual attention [2,11,30]. Recently, several methods have been proposed to detect salient regions (objects) standing out from their surroundings [10,12,18].

Past approaches for saliency computation define region uniqueness in either local or global context. *Local saliency approaches* simulate human attention mechanisms by measuring saliency as the difference between a center pixel/region and its surroundings [11,40]. The difficulty in these approaches is the selection of an appropriate neighborhood size since an object scale is unknown in advance. Therefore, local methods are efficient in predicting human eye fixation [11,33], but tend to emphasize the saliency of object boundaries rather than entire objects [43,47,61]. *Global saliency approaches* rely on color uniqueness in terms of global statistics of the image to detect salient objects [2]. For example, contrasting pixel color to all the image can be used in the spatial or the frequency domains to detect uniformly colored salient regions [2,78]. Also, color and orientation distributions can be used to compute global saliency in the image [27]. Since global methods rely on global image statistics, they are able to find regions exhibiting significant color and orientation contrast with the rest of the image. However, they are less efficient in encoding spatial information such as object location and saliency contiguity, which can cause undesired small and disconnected saliency maps. Another limitation consists in missing salient objects when they occupy a large proportion of the image.

Recently, methods combining *local* and *global* information have been proposed to enhance SOD and segmentation. For example, graph-based [35,76,77] and statistical-based methods [28,36] have been proposed for SOD. In [28], a Markov random-field model has been proposed to detect salient regions in the image. Since the graph structure is built at the pixel level, this method incurs a huge computation time. To alleviate this problem, some methods compute object saliency at the level of superpixels instead of pixels [18,35,44,77]. For example, the authors in [35] use Markov chains on graphs with nodes representing superpixels and define transient and absorbing nodes as superpixels in the center and the border of the image to formulate saliency detection. In [77], background attributes are extracted from image borders to guide SOD by graph-based manifold ranking.

* Corresponding author.
E-mail addresses: inf_tyg@yahoo.fr (I. Filali), mohandsaid.allili@uqo.ca (M.S. Allili), benblidia@gmail.com (N. Benblidia).

http://dx.doi.org/10.1016/j.image.2016.07.007
0923-5965/

Using superpixels yields generally to a huge improvement for SOD over operating on the pixel or patch level of the image [12,18,49]. One major limitation, however, remains in the selection of the optimal superpixel granularity for achieving good object detection. Indeed, too small superpixels tend generally to emphasize only some parts of the salient objects and small details of the background, whereas too large superpixels tend to produce large regions containing parts of the background [76]. To circumvent this issue, hierarchical models have been proposed to detect salient objects at different scales of the image [49,67,76]. However, in case of cluttered backgrounds or patterned small objects (e.g., flowers on grass, etc.), the above approaches can produce erroneous small salient regions. Besides, objects with small contrast with the background can cause salient objects to include large portions from the background. These undesirable effects, in turn, can decrease the performance of applications using saliency, such as object segmentation and recognition which favor connected regions with clear boundaries [68].

In this paper, we propose an approach combining supervised multi-layer graph ranking and local–global refinement for SOD. Our method is based on two main steps. In the first step, starting from a multi-scale image decomposition into superpixels, we detect roughly the location of coarse to fine parts of salient objects by optimizing an objective function on a multi-layer graph structure. In the second step, salient object boundaries are refined through an iterative process using random forests and combining region and boundary information of the image. Finally, we propose a feature weighting scheme to emphasize object saliency in case of low contrast with the background. Our approach yields generally accurate saliency maps for objects appearing at different scales and having small contrast with the background. Experiments on standard datasets containing images with complex scenes and patterns have demonstrated that our approach gives better results than recent state-of-the-art methods.

Fig. 1 illustrates our obtained saliency maps compared to six recent methods: Absorbing Markov chains (MC) [35], Dense and Sparse Reconstruction (DSR) [73], Saliency Trees (ST) [49], Saliency optimization from Robust Background Detection (RBD) [82], Extended Quantum Cuts (EQCUT) [9] and Discriminative Regional Feature Integration Approach (DRFI) [36]. Fig. 2 shows the outline of the main steps composing our algorithm for salient objet detection.

This paper is organized as follows: Section 2 presents some recent work about SOD. Section 3 presents the multi-layer graph-based saliency calculation. Section 4 presents our approach for saliency refinement using random forests. Section 5 presents some experimental results validating our approach. We end the paper with a conclusion and some future work perspectives.

2. Related work

In the last two decades, several saliency methods have been proposed [11,43]. While early approaches focused mainly on predicting human attention and eye fixation, more recent approaches are geared toward object detection [12,13]. SOD is commonly interpreted in computer vision as a process of detecting the most salient object(s) of the image [43,76]. To categorize saliency methods, different strategies can be used depending on the type of features and the prior assumptions used to help SOD [12]. Basically, two kinds of cues can be exploited for salient object detection: (1) *Intrinsic cues* refer to features such as color texture, etc., derived from a single image [13] and (2) *Extrinsic cues* refer to features such as scene depth maps [66], image annotations [12,70] or any other information extracted from multiple images sharing the same visual content [31].

In addition to visual cues, prior knowledge can be used as a high level guidance for saliency detection: (1) *Photometric priors* rely on luminance properties of the image. For example, *contrast prior* is common to all methods and assumes that salient objects have high appearance contrast with background in the spatial or the frequency domains [1,2]. *Color prior* considers that some colors (e.g., warm colors) are more attractive than other colors [79]; (2) *Spatial priors*, such as *backgroundness* [77] and *central* [71] priors make assumptions about background and salient object locations. *Contiguity prior* supposes that closer regions should share the same saliency values [77,35]; (3) *Geometrical priors* make geometrical assumptions about the objects (resp. background). For example, *compactness prior* [21] assumes that the background has a larger spread than the salient object. Recently, methods combining several of the above priors have been proposed. For example, *objectness measures* [4] propose a small number of windows likely containing objects. *Eye fixation prediction* [33,43] determines the most attractive areas for human vision. Given the models used to encode priors and image features for SOD, we can roughly find three types of methods in the literature.

2.1. Local contrast-based methods

These are generally related to eye fixation prediction by exploring the rarity of image regions with respect to local neighborhoods. Itti et al. [33] are the first to compute saliency by emphasizing the contrast of a region with regard to its neighbors. Since then, other methods based on this concept have been proposed [1,29,63]. For example, minimum conditional entropy [42] and matrix cosine similarity [63] are used to isolate local regions that cannot be predicted from their surrounding. In the same vein, conditional random fields [29,46,61] are used to predict saliency of local regions or patches with regard to their surrounding. In [37], the authors use anisotropic center-surround difference to measure the contrast between a region and its surroundings. They consider also spatial and depth priors to refine object saliency. This method yields good saliency estimation but can fail when depth information is imprecise. Local methods are generally limited since they tend to highlight object boundaries instead of entire objects. Nonetheless, local methods can be used as a good prior to initialize SOD [12,14].

2.2. Global contrast-based methods

They consider generally contrast to the whole image to determine pixel or group of pixels saliency [2,30,79]. Several

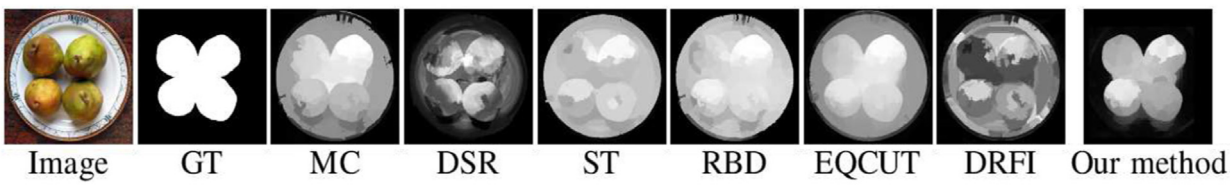

Fig. 1. Example of SOD comparing our method with recent state-of-the-art methods.

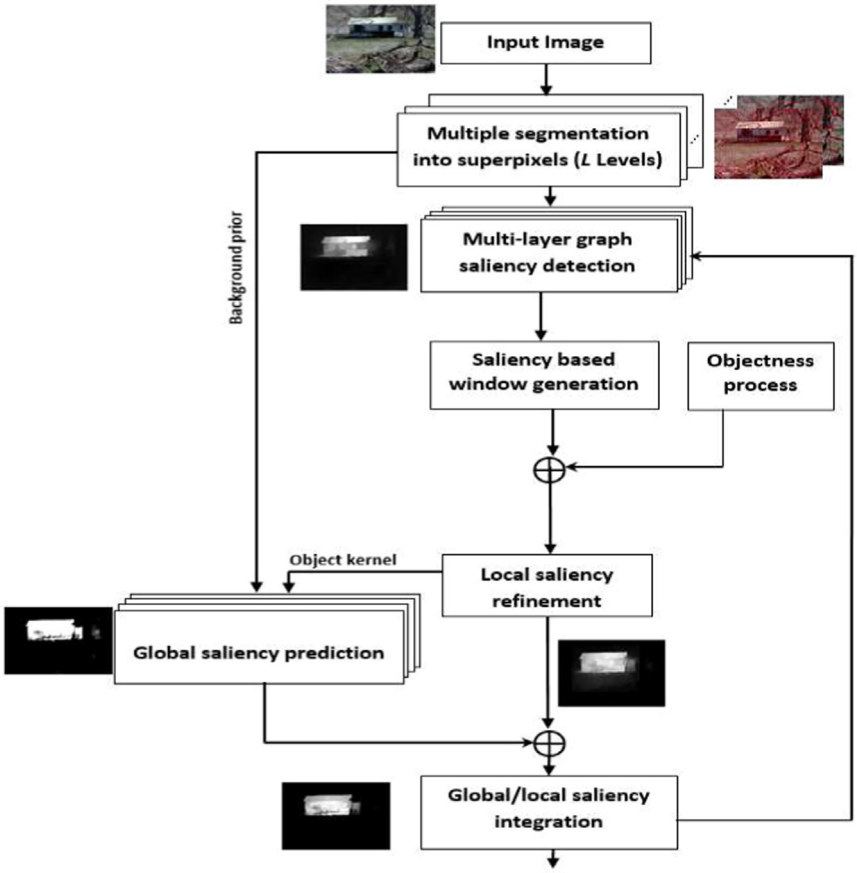

Fig. 2. The flowchart of the proposed algorithm for SOD.

approaches use this principal to detect salient regions in the image. For example, spectral-based approaches [2,30] extract uniformly colored objects in the frequency domain by subtracting the image from its low-pass filtered version. These approaches, however, are less efficient in images with cluttered backgrounds. Moreover, since they operate in the frequency domain, contiguity of salient regions is not guaranteed. Recently, other methods have proposed saliency computation using region histogram distance while enforcing global object/background compactness [18,59,78].

Contrary to local methods, global methods are better at highlighting entire regions constituting objects. However, they can produce small salient parts with obvious contrast with the rest of the image. Also, they have difficulty in distinguishing among objects and backgrounds when they contain similar color values [60]. Finally, since the object scale is unknown in advance, using a fixed decomposition of the image can cause it to miss some parts of the salient objects or merge them with the background.

2.3. Combined local–global methods

To overcome the limitations of local and global methods, several methods propose to combine global and local information of the image for SOD. Among these, we can find three main approaches:

(1) *Probabilistic methods*: Probability theory can be used to investigate the rarity of regions/patches as the least probable elements in a scene. In [32], authors use the wavelet transform and Gaussian probability density to model saliency. They also consider object contiguity and eye fixation prediction in an enhancement process. In [44], a Bayesian framework is proposed to detect salient objects by analyzing patch contrast. Similarly, [10] propose dictionary learning to compute local/global contrast of image patches. Global contrast is computed in this method as the inverse of probability of patch occurring over the entire image. Among limitations of patch-based methods are their tendency to return high-contrast edges instead of objects and object boundaries are usually not well preserved. To overcome these limitations, recent methods prefer to use superpixels instead of patches. In [81], superpixel color is modeled using a multivariate normal distribution and the Wasserstein distance is used to propagate the saliency values. In the same vein, [17,25,75] use spatial and color distributions for saliency computation. However, since these methods rely only on global image statistics, obtained object saliency can be less accurate when objects and background share similar color distributions.

(2) *Graph-based methods*: Recently, graph-based methods have emerged as an excellent tool for SOD [77,49]. In addition to the simplicity they provide for combining several image cues, graphs are efficient in encoding spatial priors such as object contiguity and location. For example, [77,49] propose graph-based methods to detect salient objects far from the image border. In [28], random walk models are used to extract salient objects on graphs. However, since the graphs are built of the image lattice, this method incurs a huge computation time. To circumvent this limitation, [35,77,82] use superpixels instead of pixels and propose to optimize functions on graphs.

To make use of multi-resolution image information, [49] use global contrast and spatial contiguity to generate initial saliency maps. Then, a region merging procedure with dynamic scale control is used to generate the so-called *saliency trees*. This method highlights salient object regions with well-defined boundaries. In [67,76], hierarchical models are proposed to estimate saliency maps at different image resolutions. Then, the different maps are averaged to build the final saliency. This method yields generally

good saliency maps, but they may assign high saliency values to isolated background regions. Note that the majority of graph-based methods use the image borders to extract the initial background. Consequently, the final saliency of objects touching the border can be degraded.

(3) *Other methods*: In addition to graph and probabilistic approaches, other methods have been proposed for SOD. In [26], global color uniqueness and some visual organization rules are combined to estimate object saliency. This method is effective in discarding background parts, but tends to highlight edges more than entire objects. In [72], *backgroundness* and *connectivity* priors are used to encode a so-called *geodesic saliency* of image patches, defined as the length of the shortest path to the background. In [34], authors use a multi-scale segmentation and Chi-square distance between region color histograms to detect salient objects. In [39], authors estimate object saliency by finding optimal linear combinations of color channels. They use relative location and color contrast between superpixels to improve the performance of saliency estimation. In [23], authors use image composition for saliency calculation. That is, each image window is tested if it can be composed of its neighbor regions. They employ also objectness, central and backgroundness priors to improve saliency estimation. The selection of window surrounding regions, however, poses a difficulty when the object scale is unknown in advance.

2.4. Discussion and contributions

Performance of SOD has increasingly improved since the introduction of superpixels and statistical methods. It remains, however, that this performance can drastically decrease in complex scenes (e.g., cluttered backgrounds, low contrast between objects and backgrounds, etc.) [14,43]. One way for improving saliency in such cases is in searching an optimal way of combining features to yield accurate saliency maps. For example, objects can appear at multiple scales and different levels of contrast with the background. Detecting and using the most discriminative features in this case can improve SOD [36]. Another way of improvement can come from exploiting region and boundary information. Object boundaries are usually characterized by high image discontinuities and, therefore, boundary information can play an important role for saliency computation. Although boundary information has been extensively used in segmentation [5,6,20], it is usually overlooked for saliency computation.

Our contribution in this paper lies basically in two main points: (1) We propose a new objective function based on multi-layer graph ranking that efficiently encode saliency estimation through multi-scale image decomposition into superpixels. The multi-layer graph structure allows to maintain consistency between salient regions obtained at different scales. It allows also to efficiently combine region and boundary features to enhance SOD accuracy, (2) we propose a procedure which refines iteratively the object boundary localization by combining global and local image information. On the one hand, this procedure uses random forests to globally separate between the object and background parts according to global image statistics. On the other hand, it uses feature relevance and combines region and boundary information for better object boundary localization.

3. Graph ranking for saliency detection

3.1. Single-layer saliency graphs

Recently, the authors in [77] have proposed a method for SOD based on semi-supervised graph manifold ranking [80]. More specifically, the image is first segmented into a set Ω composed of n regions (i.e., superpixels) obtained using color information, $\Omega = \{r_1, r_2, \ldots, r_n\}$. A weighted graph $\langle \mathcal{V}, \mathcal{E} \rangle$ is then constructed from this segmentation, where $\mathcal{V}$ is the set of nodes that correspond to regions and $\mathcal{E}$ is the set of weighted edges. A weight w_{ij} of an edge between two nodes r_i and r_j is calculated according to color similarity between the corresponding regions when they are adjacent. Otherwise, the weight is equal to zero.

Let $\phi: \Omega \rightarrow \mathbb{R}^n$ be a ranking function assigning a ranking value f_i for each region $r_i \in \Omega$. The ranking of the graph nodes $\mathbf{f} = [f_1, \ldots, f_n]^T$ starts by assigning (manually) ranks to some nodes of the graph (i.e., query nodes) which consist of superpixels in the border of the image. In other words, the four borders constitute the *backgroundness* prior as proposed in [77]. Let $\mathbf{y} = [y_1, \ldots, y_n]^T$ be an indicator vector in which $y_i = 1$ if region r_i is a query, and $y_i = 0$, otherwise. The following minimization is proposed to propagate the initial ranking (i.e., saliency values) to all the graph nodes:

$$\mathbf{f}^* = \arg\min_{\mathbf{f}} \left(\sum_{i,j=1}^{n} w_{ij} \left(f_i / \sqrt{d_i} - f_j / \sqrt{d_j} \right)^2 + \lambda \sum_{i=1}^{n} (f_i - y_i)^2 \right), \quad (1)$$

where $d_i = \sum_{k=1}^{n} w_{ik}$ and λ is a regularization constant controlling the contribution of the fidelity term to function (1).

3.2. Generalization to multi-layer saliency graphs

Since objects can appear at different scales, a fixed superpixel size can miss some parts of the objects and merge them with the background. To alleviate the problem, we propose to use L decompositions of the image into different superpixels' resolutions $\{\Omega_1, \ldots, \Omega_L\}$. Each decomposition Ω_ℓ, $\ell \in \{1, \ldots, L\}$, is generated using the SLIC algorithm [3] which produces n_ℓ superpixels $\{r_1^{(\ell)}, \ldots, r_{n_\ell}^{(\ell)}\}$. To obtain different segmentation resolutions, we use different numbers of superpixels: $n_1 < n_2 < \cdots < n_L$. We translate this on a new multi-layer graph structure as illustrated in Figs. 3 and 4. Let $\mathbf{f}^{(\ell)} = \left[f_1^{(\ell)}, f_2^{(\ell)}, \ldots, f_{n_\ell}^{(\ell)} \right]$ be the ranking of regions at level ℓ, and $\mathbf{y}^{(\ell)} = \left[y_1^{(\ell)}, \ldots, y_{n_\ell}^{(\ell)} \right]$ the indicator vector in which $y_i^{(\ell)} = 1$ if region $r_i^{(\ell)}$ is a query (i.e., non-salient) in layer Ω_ℓ, and $y_i^{(\ell)} = 0$, otherwise. We propose the following objective function generalizing (1) to multi-layer graph ranking:

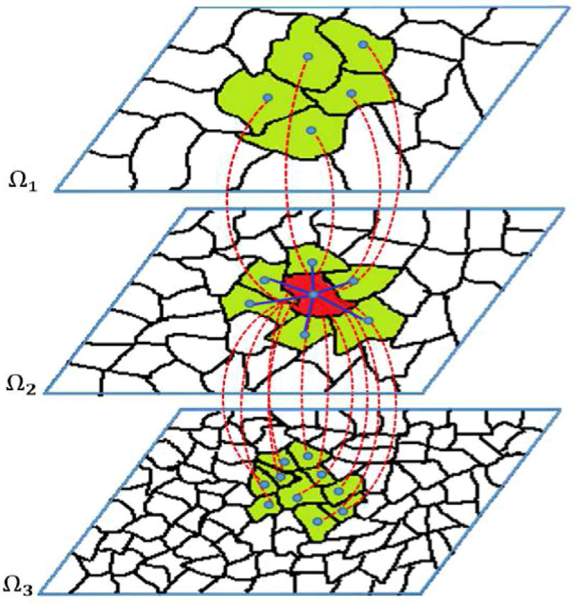

Fig. 3. Illustration of a multi-layer graph for SOD using three scales for superpixel segmentation.

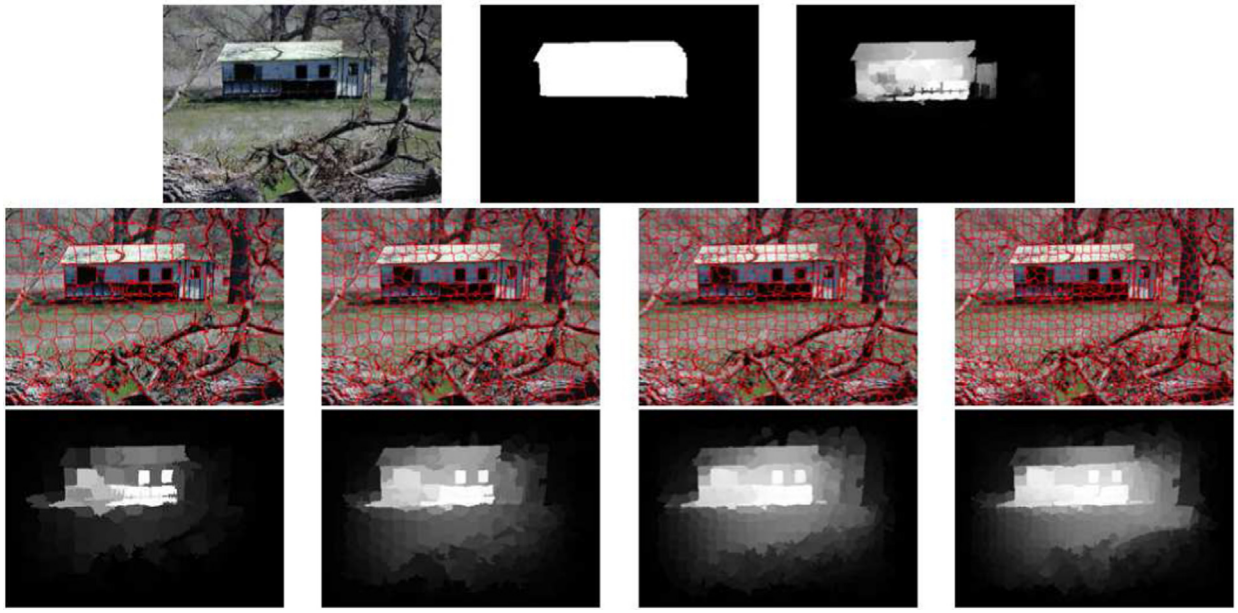

Fig. 4. Comparison of saliency maps returned by single- and multi-layer graphs. The upper row represents, from left to right, the original color image, the ground truth and our multi-layer graph saliency estimation. The middle row represents, from left to right, the segmentation of the image into 200, 400, 600 and 800 superpixels, respectively. The bottom row represents saliency maps generated by Eq. (1) using the segmentation of the same column.

$$\mathbf{f}^* = \arg\min_{\mathbf{f}} \left(\frac{1}{2} \sum_{\ell=1}^{L} \left(\sum_{i,j=1}^{n_\ell} w_{ij}^{(\ell)} \left(f_i^{(\ell)} \Big/ \sqrt{d_i^{(\ell)}} - f_j^{(\ell)} \Big/ \sqrt{d_j^{(\ell)}} \right)^2 + \sum_{m=1, m\neq\ell}^{L} \left(\sum_{i=1}^{n_\ell} \sum_{j=1}^{n_m} w_{ij}^{(\ell,m)} \left(f_i^{(\ell)} \Big/ \sqrt{\tilde{d}_i^{(\ell)}} - f_j^{(m)} \Big/ \sqrt{\tilde{d}_j^{(m)}} \right)^2 \right) + \lambda \sum_{i=1}^{n_\ell} \left(f_i^{(\ell)} - y_i^{(\ell)} \right)^2 \right), \tag{2}$$

where $\mathbf{f} = [\mathbf{f}^{(1)}, ..., \mathbf{f}^{(L)}]^T$, $d_i^{(\ell)} = \sum_{k=1}^{n_\ell} w_{ik}^{(\ell)}$ and $\tilde{d}_i^{(\ell)} = \sum_{m=1, m\neq\ell}^{L} \sum_{k=1}^{n_m} w_{ik}^{(\ell,m)}$, with $w_{ik}^{(\ell)}$ being the weight between region $r_i^{(\ell)}$ and $r_k^{(\ell)}$ belonging to the same level Ω_ℓ and $w_{ik}^{(\ell,m)}$ being the weight between region $r_i^{(\ell)}$ and $r_k^{(m)}$ belonging to levels Ω_ℓ and Ω_m, respectively. The first term of function (2) has the role of smoothing saliency between nodes of the same resolution. The second term smoothes saliency between overlapping nodes in different resolutions (nodes linked by vertical edges in Fig. 3). The third term is a regularization controlling the fidelity of the final saliency to the initial backgroundness prior. Minimization in function (2) can be solved in the same way as in function (1). First, we define the following quantities:

$$\begin{cases} D_i^{(\ell)} &= d_i^{(\ell)} + \tilde{d}_i^{(\ell)}, \\ \mathbf{W}^{(\ell)} &= \left[w_{ij}^{(\ell)} \right]; \quad i, j \in \{1, ..., n_\ell\}, \\ \mathbf{W}^{(\ell,m)} &= \left[w_{ij}^{(\ell,m)} \right]; \quad i \in \{1, ..., n_\ell\}, \ \ j \in \{1, ..., n_m\}, \end{cases} \tag{3}$$

where $\mathbf{W}^{(\ell)}$ and $\mathbf{W}^{(\ell,m)}$ are two matrices of dimensions $n_\ell \times n_\ell$ and $n_\ell \times n_m$, respectively. We then define the two matrices:

$$\mathbf{W} = \begin{pmatrix} \mathbf{W}^{(1)} & \mathbf{W}^{(1,2)} & \dots & \mathbf{W}^{(1,L)} \\ \mathbf{W}^{(2,1)} & \mathbf{W}^{(2)} & \dots & \mathbf{W}^{(2,L)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{W}^{(L,1)} & \mathbf{W}^{(L,2)} & \dots & \mathbf{W}^{(L)} \end{pmatrix} \tag{4}$$

$$\mathbf{D} = \mathrm{diag}\left[D_1^{(1)}, ..., D_{n_1}^{(1)}, ..., ..., D_1^{(L)}, ..., D_{n_L}^{(L)} \right] \tag{5}$$

Note that the elements on the diagonal of each sub-matrix $\mathbf{W}^{(\ell)}$, $\ell \in \{1, 2, ..., L\}$, are set to 0 as each node is exclusively ranked by the others (except itself). The minimum of function (2) is given by the vector $\mathbf{f}^* = (\mathbf{I} - \alpha\mathbf{L})^{-1}\mathbf{y}$, where $\alpha = 1/(1+\lambda)$, $\mathbf{L}$ is the Laplacian matrix given by $\mathbf{L} = \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{1/2}$ and $\mathbf{I}$ is the identity matrix [80]. For better performance, we consider the following ranking function using the non-normalized Laplacian matrix:

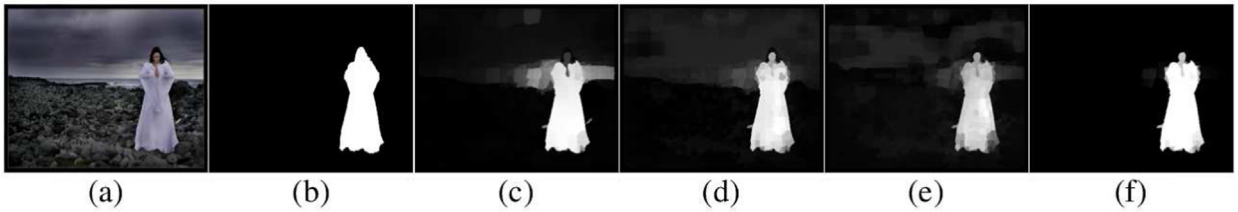

Fig. 5. Comparison between saliency maps returned after applying (c) color, (d) LBP, (e) entropy and (f) their combination. (a) and (b) represent the original image and the ground truth saliency map, respectively.

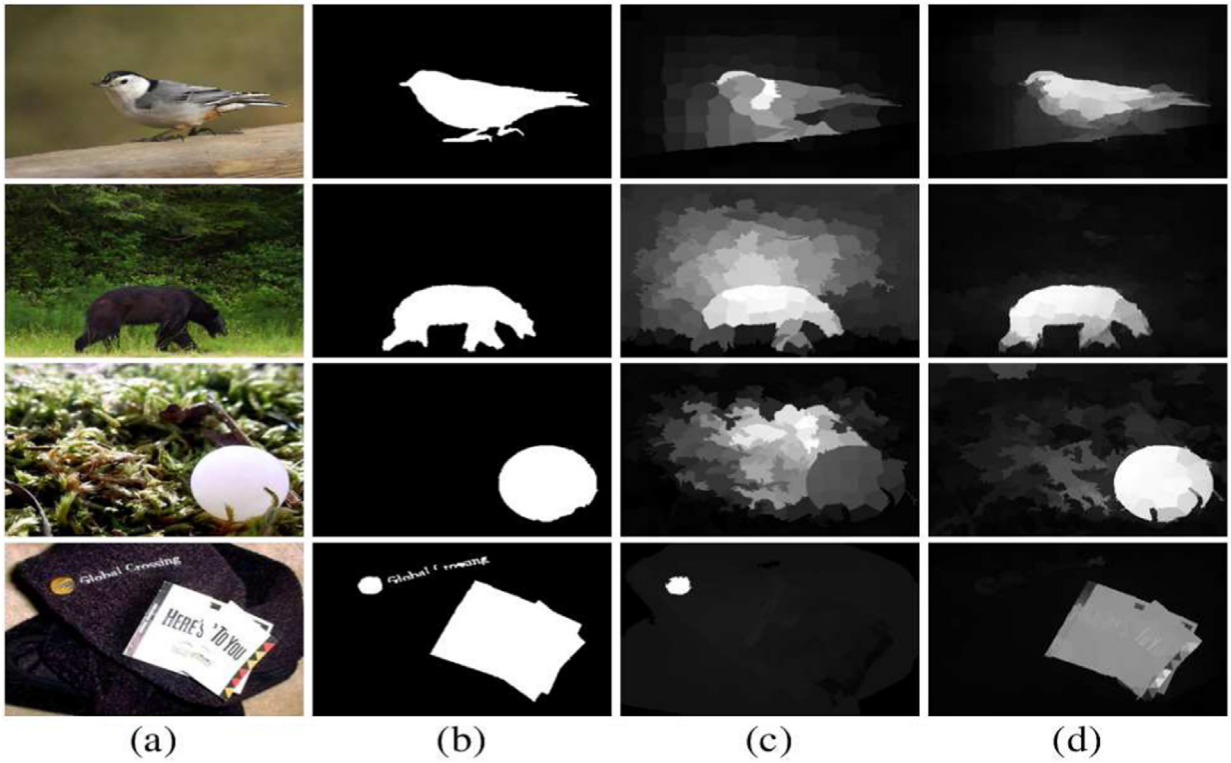

Fig. 6. Examples comparing single-layer and multi-layer graph ranking. From left to right: (a) original image, (b) ground truth, (c) single-layer graph ranking [77] and (d) multi-layer graph ranking as proposed in our method.

$\mathbf{f}^* = (\mathbf{I} - \alpha\mathbf{D}^{-1}\mathbf{W})^{-1}\mathbf{y}$ [77]. The vector $\mathbf{y} = \left[\mathbf{y}^{(1)}, \mathbf{y}^{(2)}, \ldots, \mathbf{y}^{(L)}\right]^T$ gives the initial saliency values in all resolutions.

Note that most of past methods for SOD have used only color information [12]. As can be seen in Fig. 5, texture information can benefit saliency computation where patterned objects are commonplace (e.g., grass, water, tiles of buildings, etc.). To have better description of objects, we combine region, boundary and spatial information. For region information, in each color channel k, we use color and texture features. For color, we consider the mean of each superpixel $\mathbf{c}_{i,k}^{(\ell)}$ after normalizing the color values in the range [0, 1]. For texture, we use features derived from the histograms of local binary pattern (LBP) [56] and local color entropy denoted by $\mathbf{l}_{i,k}^{(\ell)}$ and $\mathbf{e}_{i,k}^{(\ell)}$, respectively. For boundary information, we estimate the color/texture discontinuity for each superpixel by the sum of natural gradients $\mathbf{G}_{i,k}^{(\ell)}$ obtained using [20]. As will be shown in Section 4, this information is important for better localization of the object boundaries. Finally, spatial information is computed using the normalized coordinates of superpixel center of gravity denoted by $\mathbf{g}_i^{(\ell)}$. By combining the above features, we propose the following formula to calculate the weights between superpixels $r_i^{(\ell)}$ and $r_j^{(m)}$ in the multi-layer graph:

$$w_{ij}^{(\ell,m)} = \exp\left(-\gamma \sum_{k=1}^{3}\left(\xi_{c,k}\cdot\left|c_{i,k}^{(\ell)} - c_{j,k}^{(m)}\right| + \xi_{l,k}\cdot d_{\mathbb{B}}\left(\mathbf{l}_{i,k}^{(\ell)}, \mathbf{l}_{j,k}^{(m)}\right) + \xi_{e,k}\cdot d_{\mathbb{B}}\left(\mathbf{e}_{i,k}^{(\ell)}, \mathbf{e}_{j,k}^{(m)}\right)\right) + \xi_g\cdot\left\|\mathbf{g}_i^{(\ell)} - \mathbf{g}_j^{(m)}\right\|\right), \tag{6}$$

where $\xi_{c,k}$, $\xi_{l,k}$, $\xi_{e,k}$ and ξ_g are weights controlling the contribution of the different features. The weights corresponding to color and texture features are estimated by the feature relevance only at refinement process where the kernel of the object is detected beforehand. Therefore, the feature weights are set to constants in this first stage. We have assigned the highest weights to the colors since color is generally more reliable than texture. The color is weighted to 0.8 and each component of the texture to 0.2, ξ_g is a constant set to 0.02. The parameter γ controls the contribution of the boundary information in the refinement process, as will be shown in Section 4. Therefore, it is set initially to 1 for the first stage of our method. Finally, $d_{\mathbb{B}}$ denotes the Bhattacharyya distance between histograms [38]. It is defined for the LBP texture feature as follows:

$$d_{\mathbb{B}}\left(\mathbf{l}_{i,k}^{(\ell)}, \mathbf{l}_{j,k}^{(m)}\right) = 1 - \sum_{u=0}^{\rho-1}\sqrt{\left(\mathbf{l}_{i,k}^{(\ell)}(u)\cdot\mathbf{l}_{j,k}^{(m)}(u)\right)}, \tag{7}$$

where ρ is the number of histogram bins. The same formula is applied to the entropy feature. The entropy is calculated in the neighborhood of each pixel (x, y) as follows:

$$\mathbf{e}_{i,k}^{(\ell)} = -\sum_{i=0}^{\rho-1} H_k(i)\cdot\log_2(H_k(i)), \tag{8}$$

where H is the normalized local histogram for the color channel k in the $\nu \times \nu$ neighborhood around the pixel i. In our case, we set ν to 9 and ρ to 32.

For each resolution, we perform a separate saliency propagation from the four borders of the image (T: top, D: down, R: right and L: left) by choosing a proper initialization of y in Eq. (2). The saliency at superpixel $r_i^{(\ell)}$ can then be computed using the formula $s_{*,i}^{(\ell)} = 1 - f_{*,i}^{(\ell)}$ where $* \in \{top, down, right, left\}$. The obtained saliency maps S_{top}, S_{down}, S_{right} and S_{left} are then combined as follows: $S_0 = \mathbf{S}_{top} \bigcirc \mathbf{S}_{down} \bigcirc \mathbf{S}_{right} \bigcirc \mathbf{S}_{left}$, where $\bigcirc$designates the Hadamard product between matrices. We then perform a second propagation process from the most salient elements extracted from S_0 to the rest of the image to obtain a second saliency map S_1.

Fig. 4 gives an illustration on how the superpixels granularity has a direct impact on the generated saliency map. Indeed, low

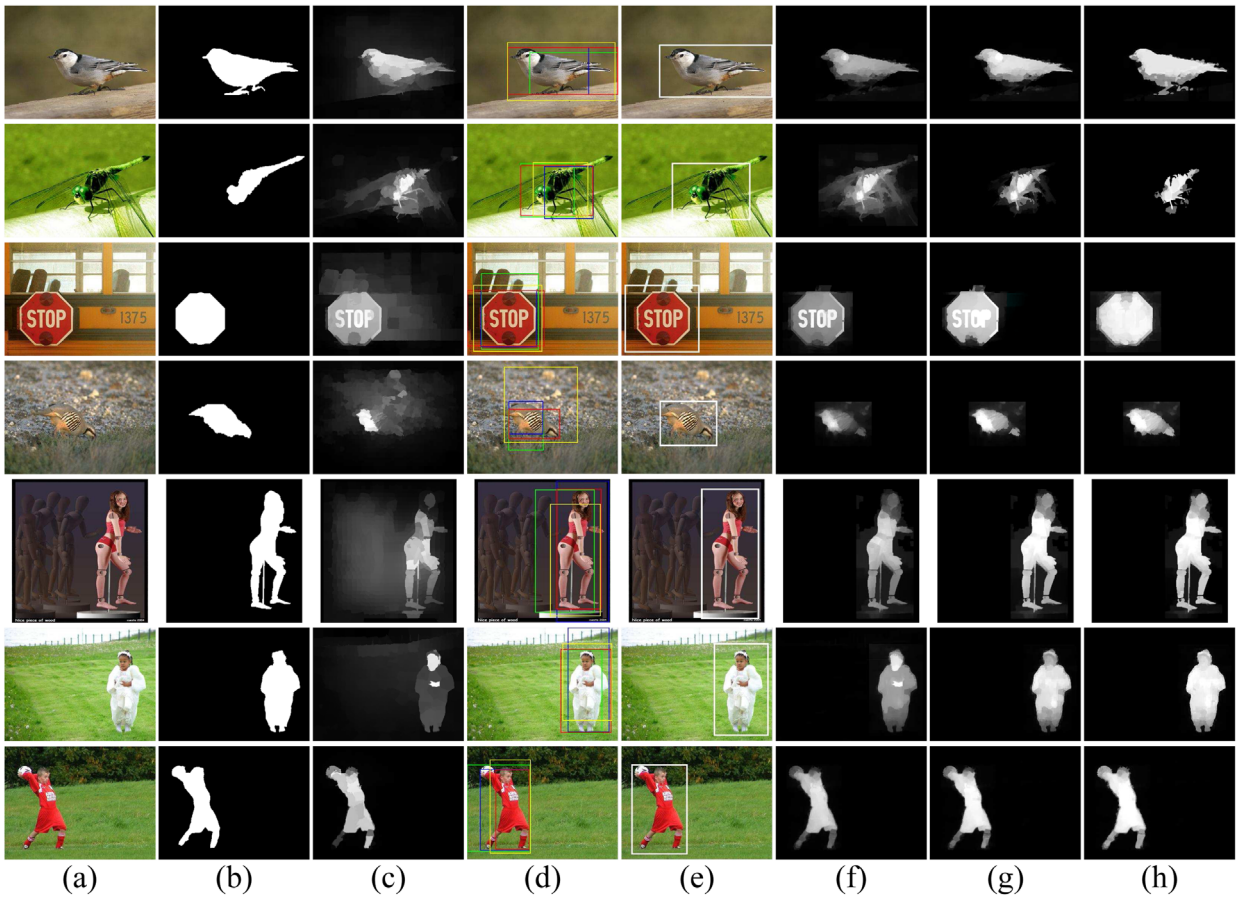

Fig. 7. Examples of window-based refinement: (a) the input image, (b) the ground truth, (c) saliency map obtained using Eq. (2), (d) positions of the windows ($\mathcal{W}$): the yellow rectangle corresponds to W and the others to $B_{(i)}$, $i = 1, 2, 3$, (e) position of the final window W', (f)–(h) are saliency maps obtained using window refinement process: (f) without feature relevance and boundary information, (g) with feature relevance and without boundary information and (h) with feature relevance and boundary information.

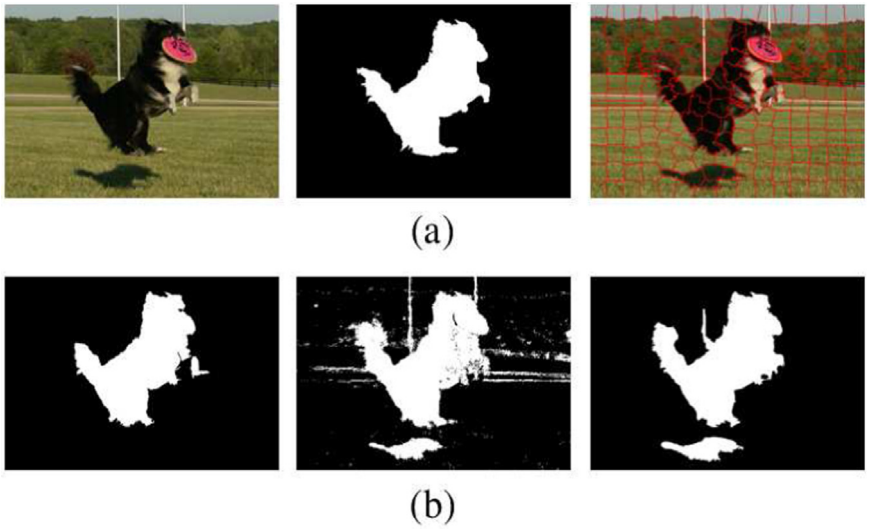

Fig. 8. Example of random-forest-based saliency calculation using the left border of the image as background prior. From left to right, (a) represent the input image, the ground truth and the segmentation into 200 superpixels, respectively, (b) represent the obtained object kernel, the result of random forest prediction and the map of the unified labels, respectively.

granularity segmentation tends generally to return a reduced portions of objects, while high granularity tends to return objects and their surrounding parts in the background. Albeit object surrounding can carry important information about the object context [26,69], it might be undesirable for applications such as object recognition [68], pose estimation [48,58] and image editing [16]. Fig. 6 shows examples comparing SOD using multi-layer versus single-layer graph ranking. We can note that the salient object boundaries are more accurately detected using our approach based on multi-layer graphs than single-layer ones.

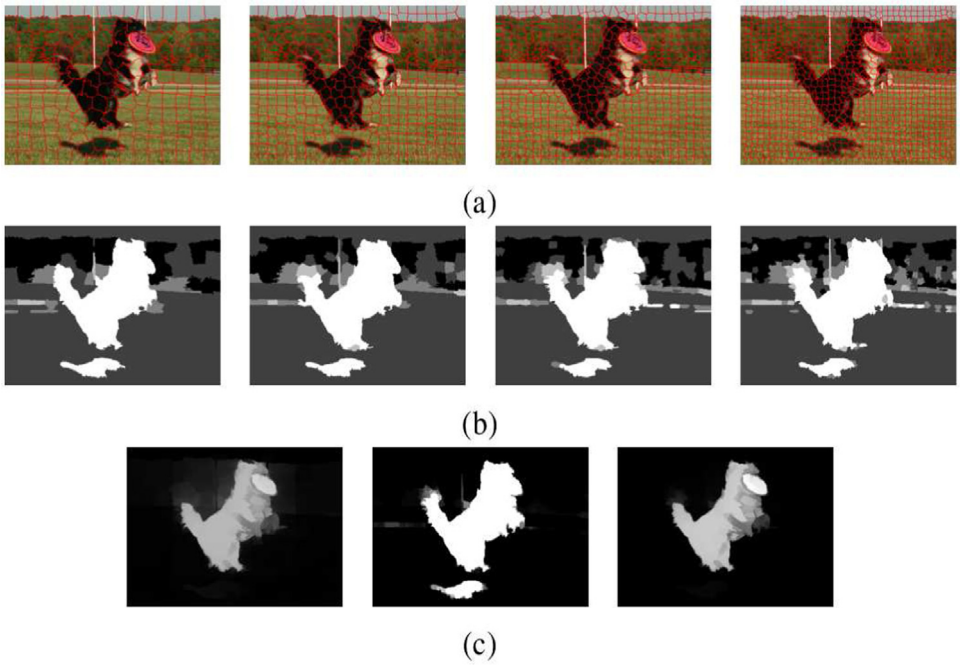

Fig. 9. Illustration of the main steps of the random-forest-based saliency refinement on the example in Fig. 8. From left to right: (a) represents image segmentations using 200, 400, 600 and 800 superpixels, respectively, (b) represents saliency maps using random forests and the segmentations in row (a), (c) the first image represents refinement results using FR and boundary information, second image represents the fusion of the RF-based maps obtained in (b) and the final image represents the combination of maps using Eq. (18).

4. Saliency refinement using feature relevance and random forests

In our approach, we aim to obtain a refined object saliency with clear boundaries. Three improvements are operated on the obtained saliency in the first stage. The first improvement consists of using a saliency-oriented window generation combined and objectness estimation to narrow the space of search for SOD. The second improvement consists of introducing a procedure using random forests to endure that the final object is consistent with global foreground/background statistics of the image. The third improvement consists of introducing a feature relevance scheme combining region and boundary information to enhance object boundary localization. These steps are detailed in sections below.

4.1. Spatial saliency refinement

For more precise saliency estimation, we use a refinement procedure to narrow the area containing the salient object. This aims at fitting an appropriate window surrounding the salient object where the final salient object is located. Let S_1 denote the saliency obtained in the first stage by minimizing function (2). To search for the object window, we first extract the object kernel made of parts having high saliency values. This is achieved by carrying out a binary segmentation on S_1 using finite Gaussian mixture models (GMMs) [54].

Let $C_1, C_2, \ldots, C_K$ be K groups generated using GMM-based clustering and sorted in an ascending order of their mean parameter values $m_1, m_2, \ldots m_K$. The parameter K representing the number of mixture components which are estimated using the minimum message length (MML) method [7,54]. Let $w_1, w_2, \ldots, w_K$ be the weights of these components. Since brighter areas in the saliency map are more likely to belong to the salient object, we form an initial background and foreground (object kernel) by separating the mixture components into 2 groups: $C_b = \{C_1, C_2, \ldots, C_k\}$ and $C_f = \{C_{k+1}, C_{k+2}, \ldots, C_K\}$, where k is defined as follows:

$$k = \arg\min_{h}\left(\sum_{i=1}^{h} w_i > \tau\right), \tag{9}$$

where τ is a threshold defined in the range $[0, 1]$ determining the confidence level at which the object kernel is extracted (usually $\tau = 0.95$).

Let *Seg* be the binary mask obtained after classification of pixels into classes C_f and C_b, respectively, such that $Seg(x, y) = 1$ if $S_1(x, y) \in C_f$ and $Seg(x, y) = 0$, otherwise. To calculate the center of the window, called *anchor point*, we use the following formula:

$$(x_0, y_0) = \frac{1}{Z}\sum_{x=1}^{m_1}\sum_{y=1}^{m_2}(x, y) \times \log\big(1 + Seg(x, y) \times S_1(x, y)\big), \tag{10}$$

where $Z = \sum_{x=1}^{m_1}\sum_{y=1}^{m_2}\log\Big(1 + Seg(x, y) \times S_1(x, y)\Big)$ and (m_1, m_2) are the dimensions of the image, $S_1(x, y) \in [0, 255]$. From the anchor point, we define a rectangular window W with high plausibility to include the salient object (i.e., objectness-like measure). The window extent is defined as $\mathcal{X}_{\delta_1} = [x_0 - \delta_1, x_0 + \delta_1]$ and $\mathcal{Y}_{\delta_2} = [y_0 - \delta_2, y_0 + \delta_2]$, with δ_1 and δ_2 obtained using the following formula:

$$(\delta_1, \delta_2) = \arg\max_{\delta_1,\delta_2}\left\{\frac{1}{Z'}\sum_{x\in\mathcal{X}_{\delta_1}}\sum_{y\in\mathcal{Y}_{\delta_2}}\log\big(1 + S(x, y)\big)\right\} \leq 1 - \eta, \tag{11}$$

where $Z' = \sum_{x=1}^{m_1}\sum_{y=1}^{m_2}\log\Big(1 + S_1(x, y)\Big)$ and η is a threshold set experimentally to 0.01. Basically, Eq. (11) yields a window surrounding the most salient parts of the image.

The saliency estimation inside W may enhance significantly the results. However, it is fully dependent on the initial saliency map S_1. If S_1 misses some parts of the salient object the window position will be affected. Conversely, if S_1 contains false positives, it can include parts of the background inside the salient object. For example, in the illustration shown in Fig. 7, our generated window (presented with yellow color) misses some details of the salient

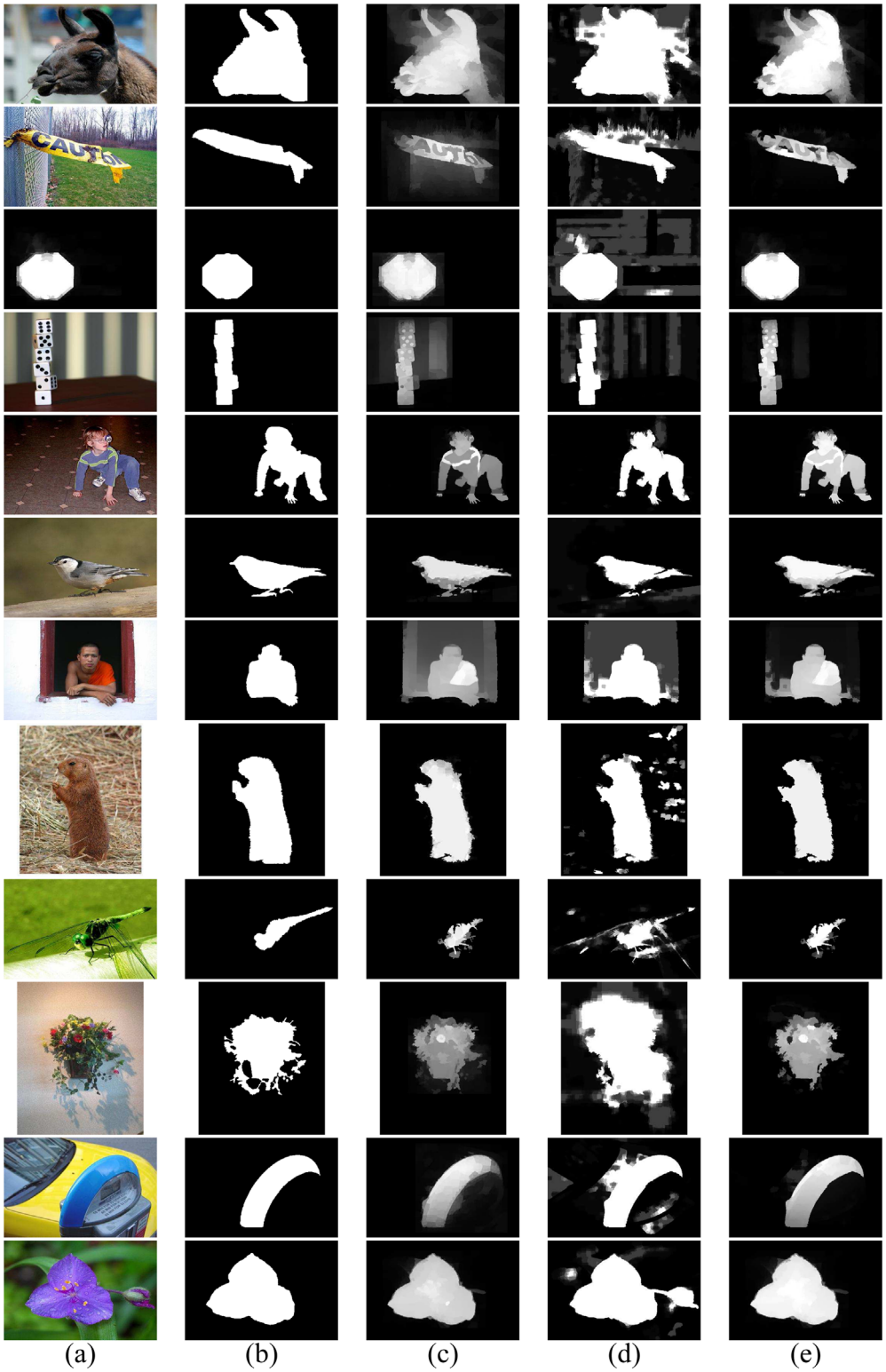

Fig. 10. Examples illustrating saliency refinement using our method: (a) the input image, (b) the ground truth, (c) saliency map obtained using our windowing process, (d) saliency map obtained using random forests saliency prediction, and (e) the final saliency map.

objects in rows 5, 6, 7 and it encompasses a large area of the salient object in row 4. To alleviate these issues, we use another objectness measure independent of S_1 calculated using [83]. This method produces n top-scored boxes, among which we choose $m \ll n$ having the greatest overlap with our window W. Experimentally, we set $n=20$ and $m=3$.

Let $B_1, B_2, \ldots, B_n$ be the n top-scored generated boxes. For each box B_i, we compute its spatial overlapping score with W as $O_i = (B_i \cap W)/(B_i \cup W)$. By choosing the m most overlapping boxes: $B_{(1)}, \ldots, B_{(m)}$, we increase the probability of obtaining a new window encompassing the salient object. By considering the set of windows $\mathcal{W} = \{W, B_{(1)}, \ldots, B_{(m)}\}$, the final window W' is made of

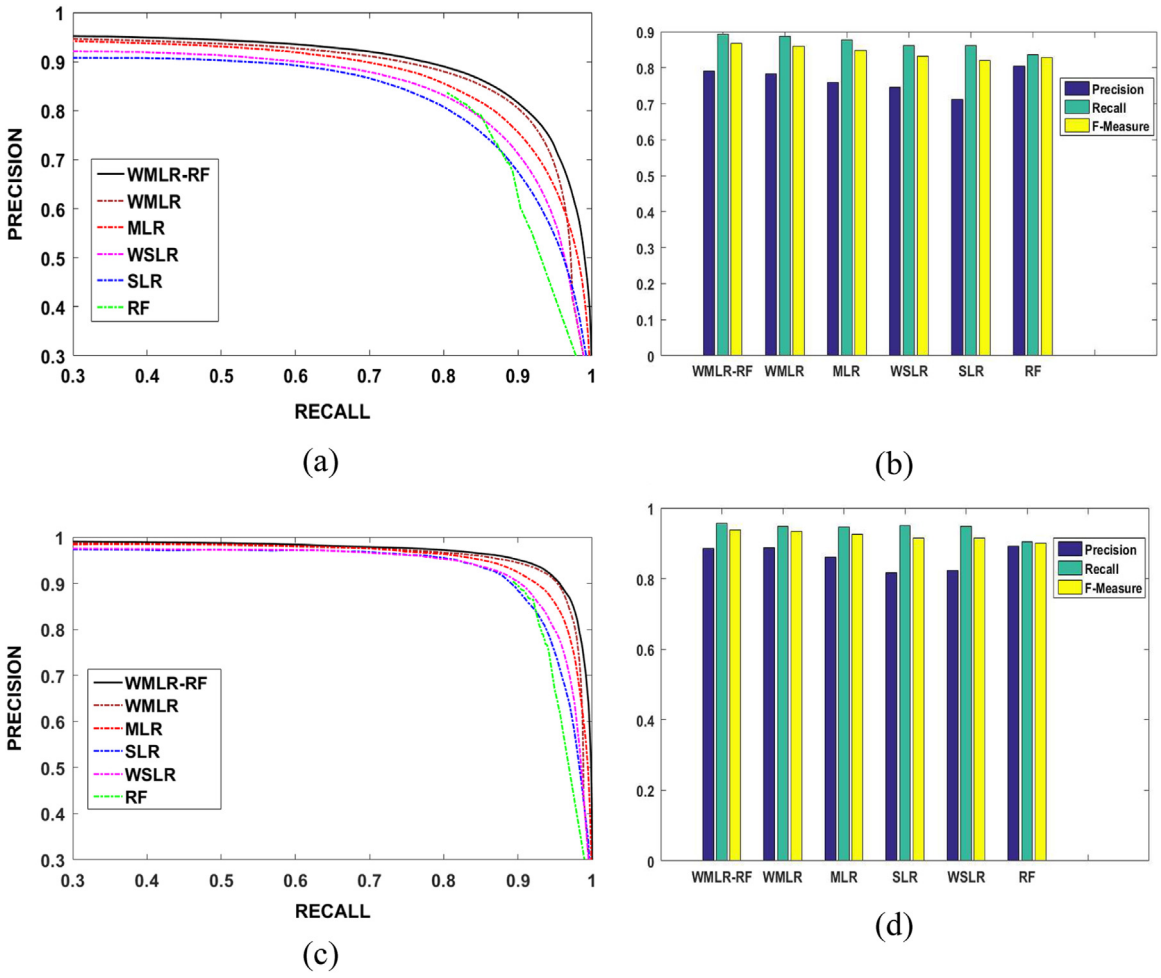

Fig. 11. Comparative results for the SLR, WSLR, MLR, WMLR, RF and WMLR-RF methods using (a, b) the MSRA-B dataset and (c, d) the ASD dataset. For each dataset, the left graph represents the obtained precision-recall curves and the right graph represents the best values obtained for R, P and F_μ measures.

the bounding box surrounding the set of points belonging to at least three elements of $\mathcal{W}$. Fig. 7 gives some examples of window calculation. Clearly, the salient objects have been narrowed with high accuracy using the proposed approach.

4.2. Saliency refinement using feature relevance and boundary information

Once the window likely containing the salient object is generated, we run another time the multi-layer graph ranking using Eq. (2) inside the window. We use the window border as the background prior to initialize the graph ranking. To better localize the object boundaries, we introduce feature relevance (FR) in the graph weights of Eq. (6) through the parameters $\xi_{c,k}$, $\xi_{l,k}$ and $\xi_{e,k}$, $k = \{1, 2, 3\}$ and γ. FR is aimed for assigning more importance to features having better discrimination between the salient object and the background. Note that since we have the object kernel obtained by the segmentation operated by Eq. (9) and the background prior, FR can be estimated in a supervised fashion.

For a given feature, let p and q be the normalized histograms for this feature in the salient object and the background, respectively. We suppose that each feature is quantized into 256 bins, and let $p(i)$ and $q(i)$ be the normalized frequencies of the i-th bin. The feature discrimination power is reflected by the degree of overlapping between p and q, which can be expressed using the following formula [19]:

$$V = \frac{var(F; (p+q)/2)}{var(F; p) + var(F; q))}, \tag{12}$$

where $var(F; p)$, $var(F; q)$ and $var(F; (p+q)/2)$ represent the variance of the function F obtained with respect to the distribution p, q and $(p+q)/2$, respectively. The value of the function F for the i-th bin is given by the log-ratio:

$$F(i) = \log\left(\frac{\max(p(i), \epsilon)}{\max(q(i), \epsilon)}\right) \tag{13}$$

where ϵ a small value that prevents null frequencies. We can demonstrate that the function V in (12) takes its values in the interval $[0.5, +\infty[$. To normalize FR into the interval $[0, 1]$, we use the following lenient function:

$$\xi = 1 - \exp(-\delta|V - 0.5|), \tag{14}$$

where the factor δ controls the sensitivity of relevance calculation. Increasing the value for this factor will increase the number of considered features as relevant and vice versa.

Finally, we introduce boundary information in the graph weights through the parameter γ of function (6). Since we perform a saliency diffusion from the window borders inward, object boundaries are likely encountered where image has high discontinuities. We use [20] to estimate the natural gradient of images. Suppose the sum of gradients in a superpixel $r_i^{(\ell)}$ is given by $\mathbf{G}_i^{(\ell)}$. We propose to set the value of γ to $(\mathbf{G}_i^{(\ell)} + \mathbf{G}_j^{(m)})/2$ for the inward diffusion operated by Eq. (2). As the salient object

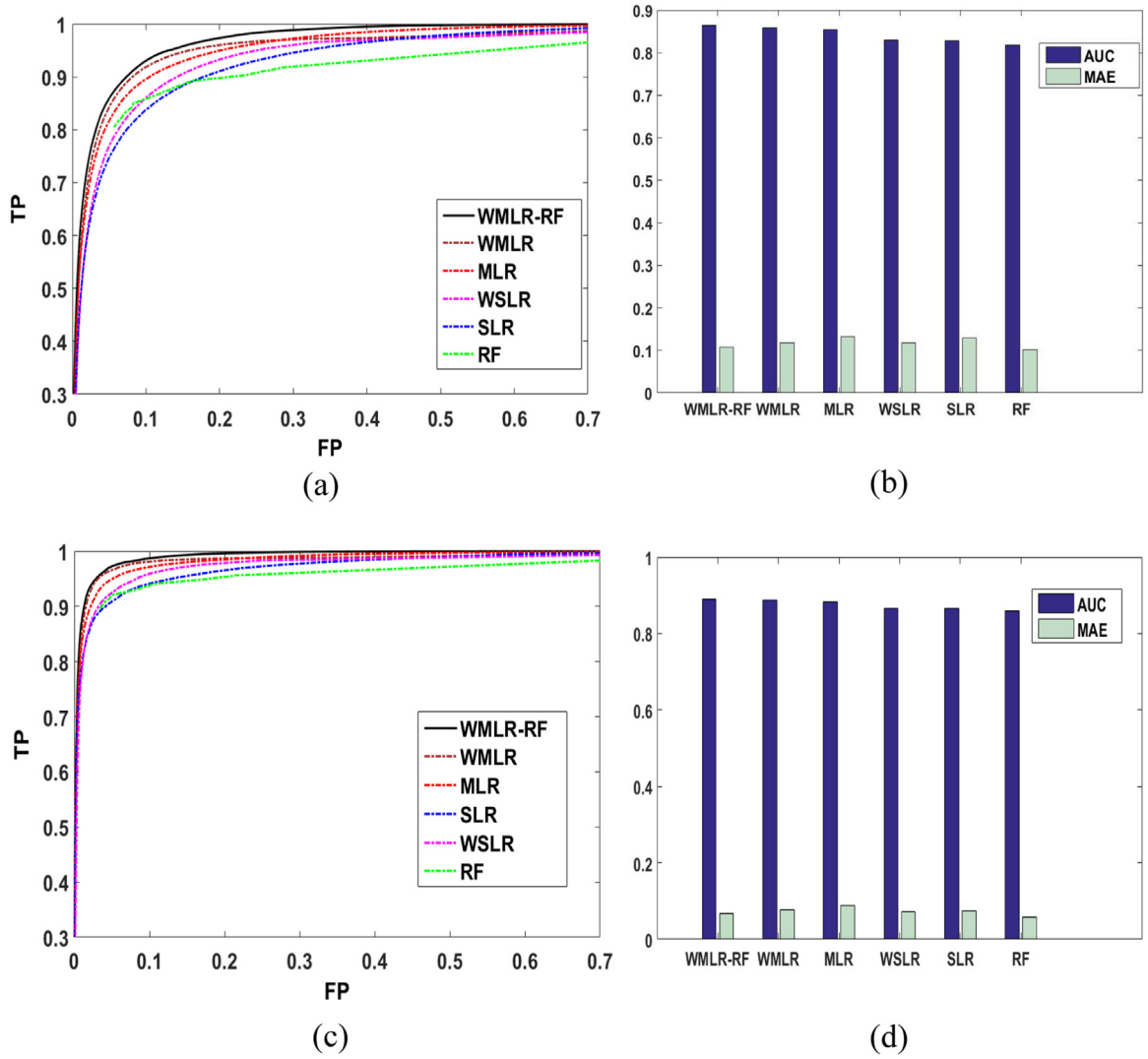

Fig. 12. Comparative results for the SLR, WSLR, MLR, WMLR, RF and WMLR-RF methods using (a, b) the MSRA-B dataset and (c, d) the ASD dataset. For each dataset, the left graph represents the obtained ROC curves and the right graph represents the AUC and MAE measures.

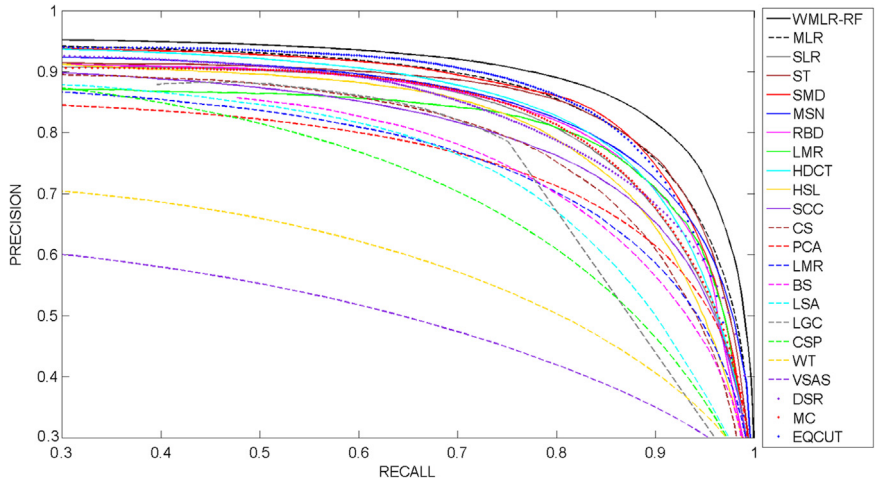

Fig. 13. Obtained *precision–recall* curves for 21 compared methods using MSRA-B dataset.

boundaries have high gradient values, the weight between two adjacent superpixels located on the vicinity of object boundaries will tend to 0.

Once all the weights and feature relevance are estimated, we apply Eq. (2) on the sub-image delimited by the window W' to calculate a saliency map denoted by S_2. Fig. 7 shows some

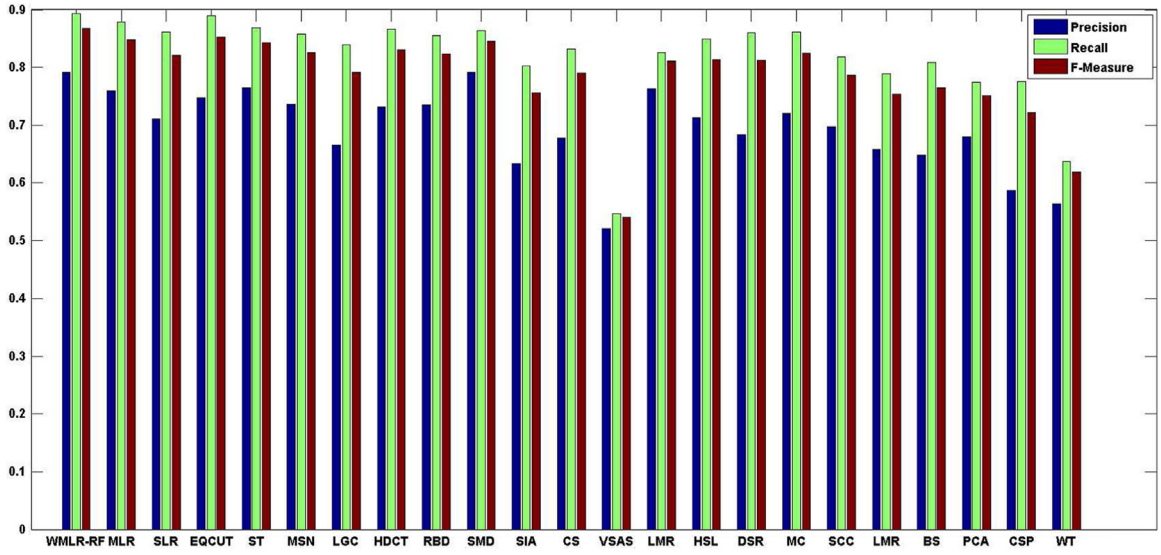

Fig. 14. Obtained values for R, P and F_μ measures for 21 compared methods using MSRA-B dataset.

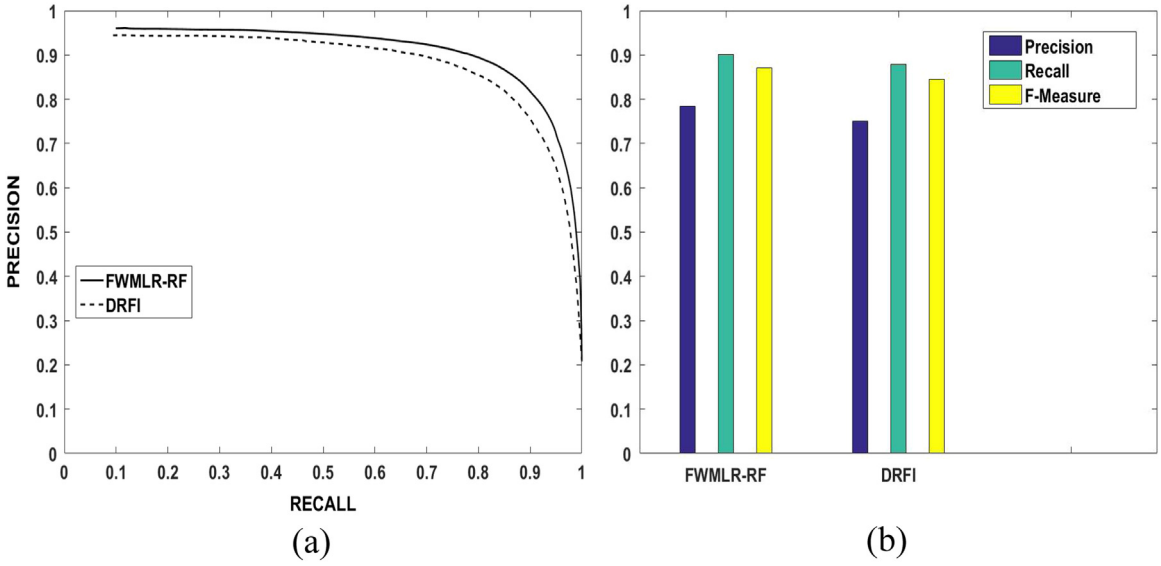

Fig. 15. Comparative results between our saliency detection method and DRFI on 2000 images of MSRA-B dataset: (a) Precision–Recall curves and (b) best F-measure and it corresponding Precision and Recall.

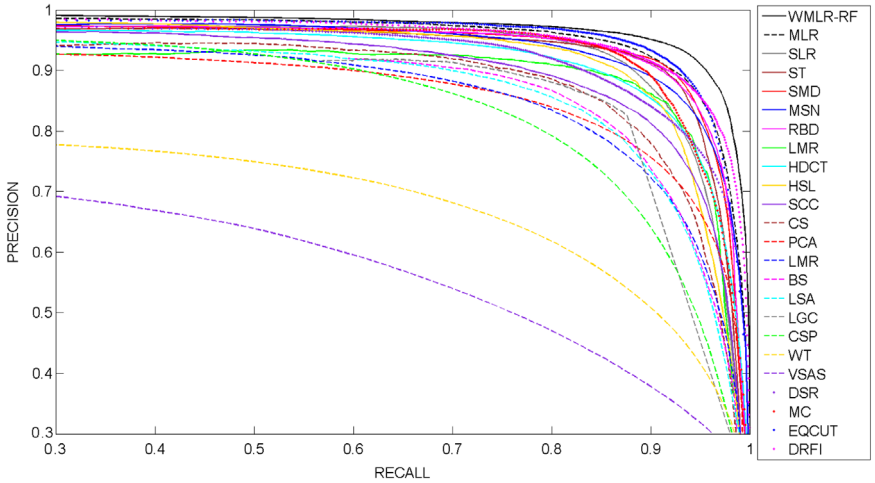

Fig. 16. Obtained precision–recall curves for the 22 compared methods using ASD dataset.

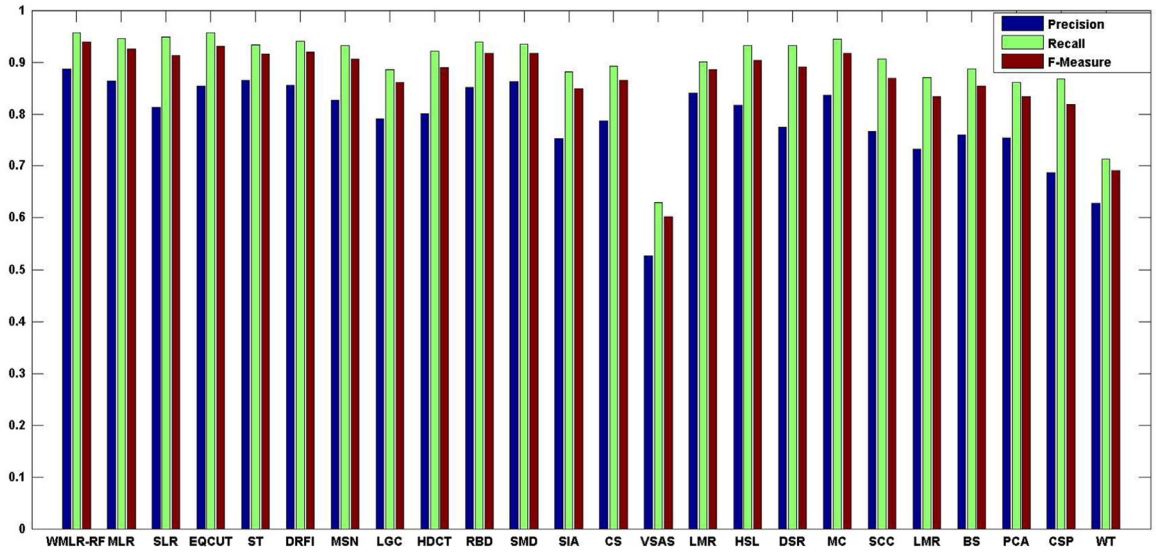

Fig. 17. Obtained values for R, P and F_μ measures for 22 compared methods using ASD dataset.

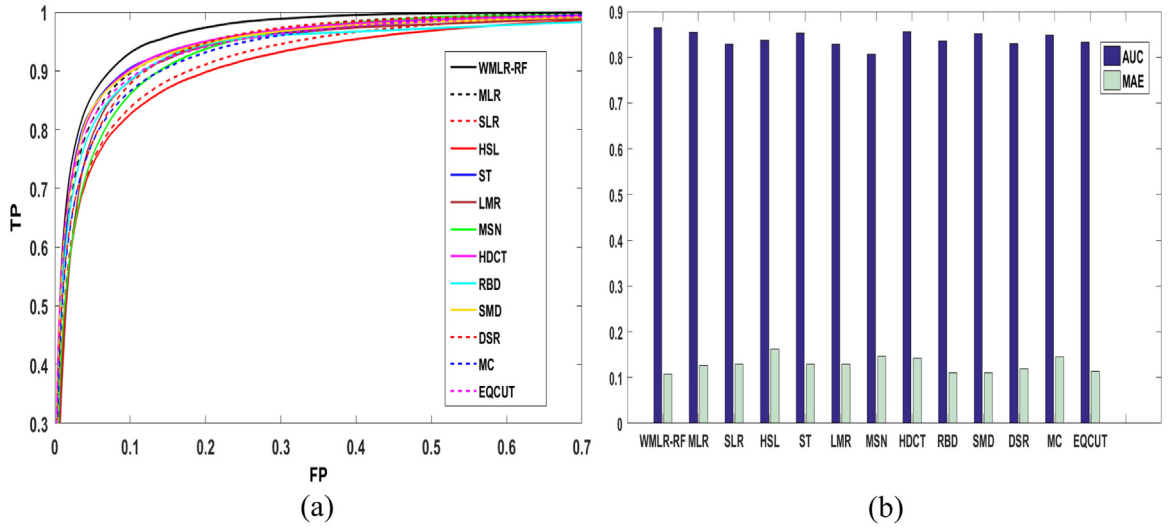

Fig. 18. Comparative results between our saliency detection method and eleven best salient object detection methods on MSRA dataset: (a) Obtained ROC curves and (b) AUC and MAE rates.

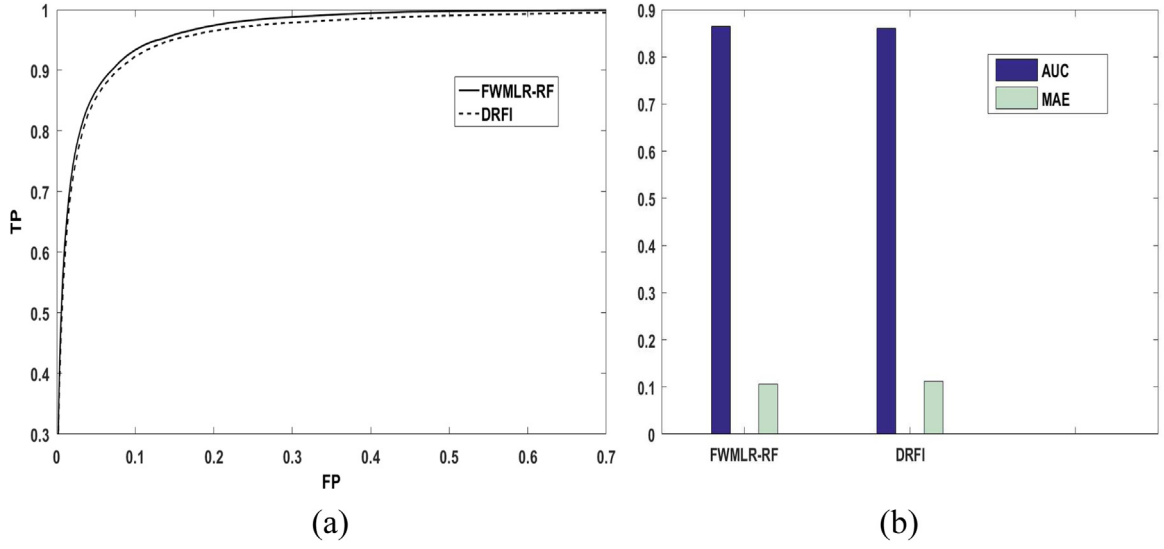

Fig. 19. Comparative results between our saliency detection method and DRFI on 2000 images of MSRA-B dataset: (a) obtained ROC curves and (b) AUC and MAE rates.

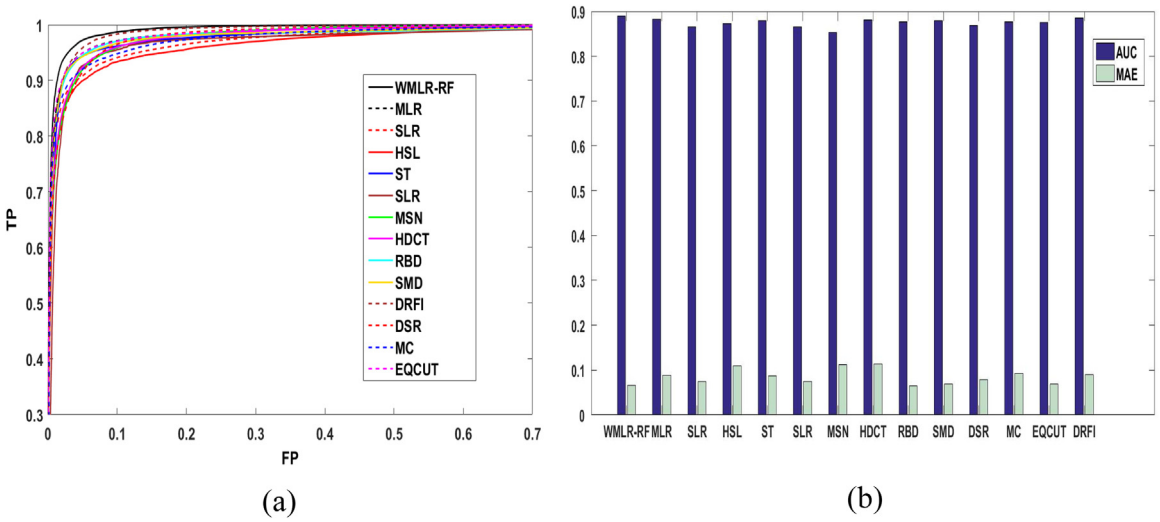

Fig. 20. Comparative results between our saliency detection method and twelve best salient object detection methods on ASD dataset: (a) obtained ROC curves and (b) AUC and MAE rates.

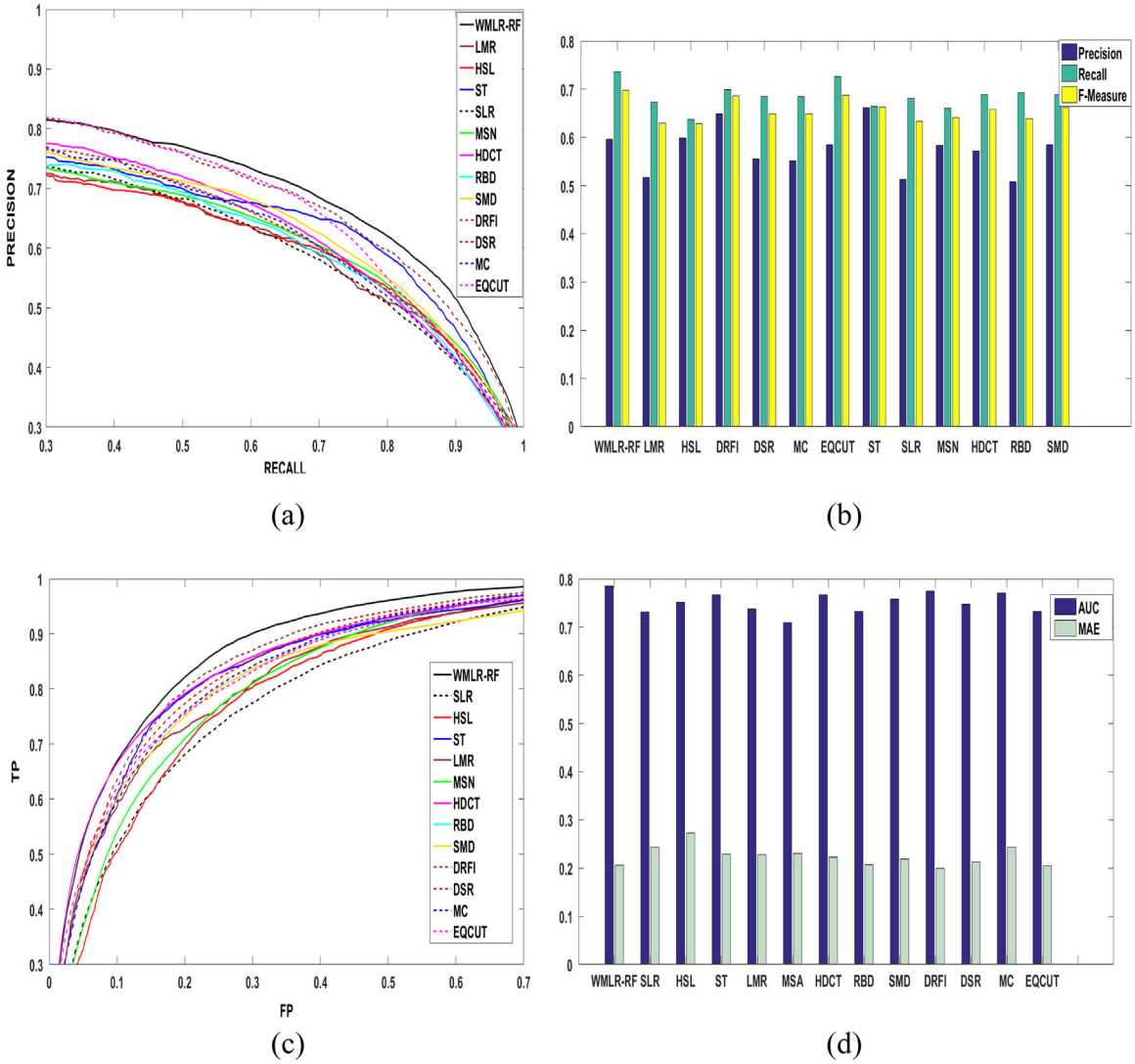

Fig. 21. Comparative results between our saliency detection method and twelve best salient object detection methods on SOD dataset: (a) PR curves, (b) precision, recall and F_μ rates, (c) obtained ROC curves and (d) AUC and MAE rates.

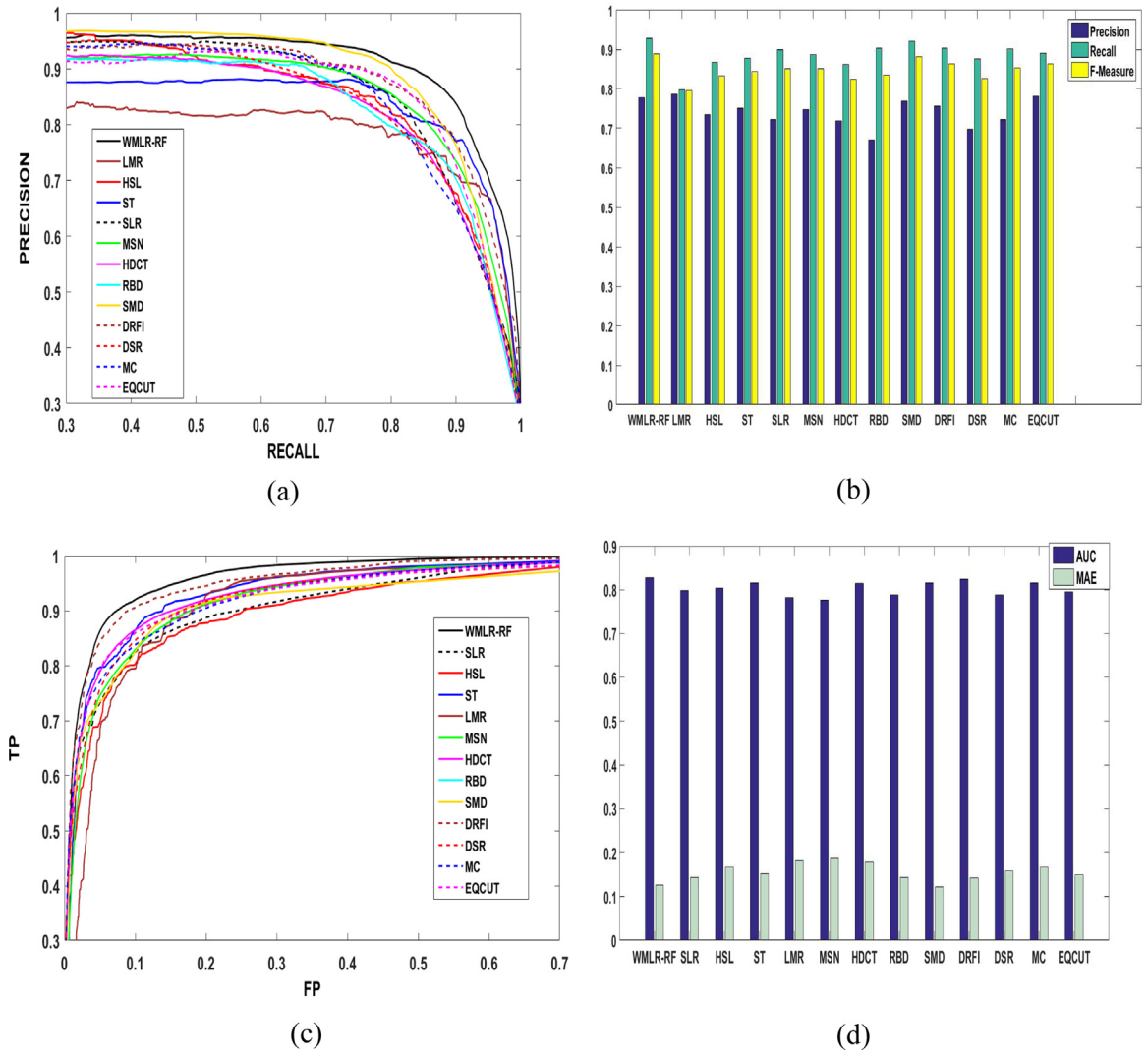

Fig. 22. Comparative results between our saliency detection method and twelve best salient object detection methods on SED1 dataset: (a) PR curves, (b) precision, recall and F_μ rates, (c) obtained ROC curves and (d) AUC and MAE rates.

examples where saliency is locally enhanced using our multi-size windowing procedure. We can note that the inclusion of the gradient and feature relevance allows to discard more efficiently the background while ensuring better object boundary localization.

4.3. Saliency refinement based on random forests

To uncover clearer object boundaries, we use supervised learning based on random forests (RF) [15] to maintain a saliency map inline with the global image statistics. A RF classifier is trained for each resolution level using superpixels constituting the object kernel obtained from segmentation and superpixels constituting the background of the level. To reduce false saliency predictions at the vicinity of the object boundaries, we propose a procedure that assigns RF-based saliency values for superpixels while taking into account the level of contrast between the object and the background.

First, using color and texture features, we model the object kernel and background parts using two multivariate Gaussian mixture models (GMMs): M_{obj} and M_{bck}. Let K_{obj} and K_{bck} be the number of components constituting the two models, which are estimated using the MML principle [54]. To estimate the level of contrast d between the object and the background, we calculate the following Kullback Leibler divergence between the components of M_{obj} and M_{bck}:

$$d = \min_{i,j}\left(\mathrm{tr}\left(\Sigma_j^{-1}\Sigma_i\right) + (\mu_j - \mu_i)^T\Sigma_j^{-1}(\mu_j - \mu_i) + \ln\left(\frac{|\Sigma_i|}{|\Sigma_j|}\right)\right), \tag{15}$$

where $i \in \{1, \ldots, K_{obj}\}$ and $j \in \{1, \ldots, K_{bck}\}$. The parameters (μ_i, Σ_i) and (μ_j, Σ_j) are the mean vector and the covariance matrix of the i-th and j-th Gaussian of the models M_{obj} and M_{bck}, respectively. Using the contrast level d, we compute a dynamic threshold that allows to affect a unified label for each superpixel $r_i^{(\ell)}$. For each resolution level Ω_ℓ, $\ell \in \{1, \ldots, L\}$, we use color and texture features to train random forest (RF) classifiers to separate each border of the image from the object kernel. Each RF classifier is constituted of three trees and each tree is trained by selecting randomly 50% of the available data and three features from a set of 9 features. By carrying out saliency prediction for each pixel, let $p_i^{(\ell)}$ be the percentage of pixels in each superpixel $r_i^{(\ell)}$ classified as salient object. The unified saliency label for $r_i^{(\ell)}$ is determined as follows:

$$s_i^{(\ell)} = \begin{cases} 1 & \text{if } p_i^{(\ell)} \geq \exp(-\beta d) \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

The parameter $\beta > 0$ controls the sensitivity of unifying label generation in each superpixel (usually $\beta \approx 0.1$). The final labelling of the superpixel depends on the similarity between foreground and background distributions. The higher the values of the contrast d, the more confidence the algorithm gives to the RF prediction.

Fig. 8 shows an example of saliency prediction using the left border of the image as a background prior. Note that unifying label generation increases the quality of the saliency estimation where noise is considerably reduced. The saliency map T_ℓ of the resolution $\ell \in \{1, \ldots, L\}$ is then calculated by averaging the saliency maps generated by the four borders. Finally, we combine the saliency maps of all resolutions by factorizing the saliency maps T_ℓ as follows (see Fig. 9):

$$T_f = T_1 \times T_2 \times \cdots \times T_L. \quad (17)$$

Fig. 9 shows an example of our multi-scaled random forest process. Multi-scaled random forest algorithm returns high values to salient regions. The salient object stands out more effectively with the increased recall rate. However, we note that the salient regions returned contain generally some false positives that reduce significatively the precision of detection. In the other hand, our graph based algorithm returns more restricted salient regions that increase the precision rate. However, the salient object is less highlighted and the recall may decrease in some cases. In order to reach jointly high precision and recall rates, we integrate the two saliency maps according to the following formula:

$$S_3 = [\omega * T_f + (1 - \omega) * S_2] * S_2; \qquad \omega \in [0, 1] \quad (18)$$

with ω being a parameter balancing the contribution of T_f and S_2 to the final saliency map. Usually, we assign more importance to S_2 by setting $\omega < 0.5$ (we set experimentally ω to 0.33). This allows better highlighting of salient objects than in S_2 while discarding false foregrounds produced by T_f. The multiplication by S_2 allows for further elimination of false foregrounds without affecting high foreground saliency values. Finally, the main steps of the proposed SOD algorithm are summarized in Algorithm 1. The parameter $iter_{max}$ gives the maximum number of iterations involved in the refinement process (usually $iter_{max} = 2$). Note that higher values of

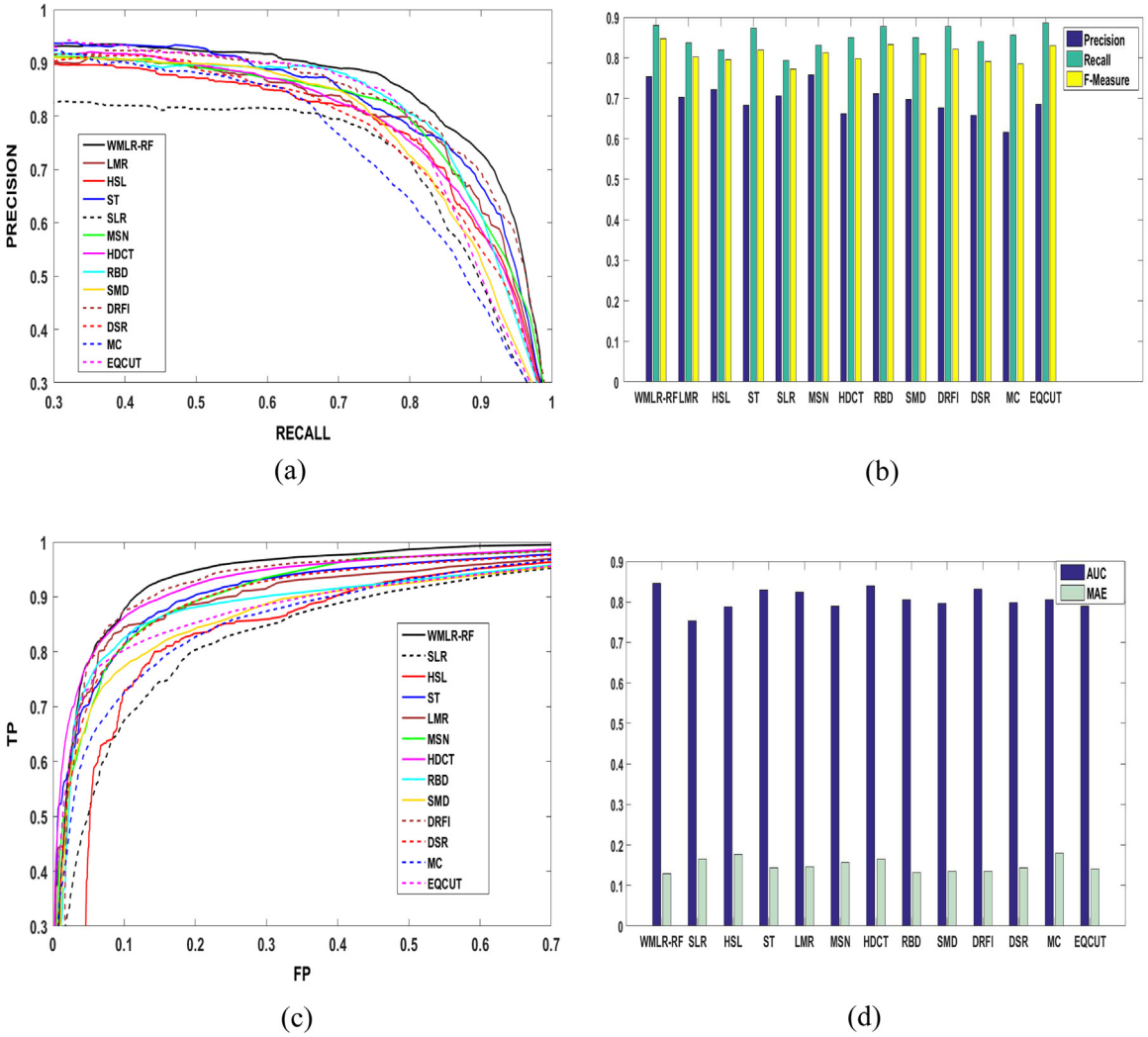

Fig. 23. Comparative results between our saliency detection method and twelve best salient object detection methods on SED2 dataset: (a) PR curves, (b) precision, recall and F_μ rates, (c) obtained ROC curves and (d) AUC and MAE rates.

$iter_{max}$ can make the algorithm degenerate since the window size progressively shrinks to capture only local parts of objects. To avoid such situations to occur, we devised a more conservative scheme when augmenting the number of iterations by setting the parameters α and η controlling Eqs. (2) and (11), respectively, to $\alpha = 9 \times 10^{-iter}$ and $\eta = 100^{-iter}$. Fig. 10 shows an example of saliency computation using Eq. (18). We can see that the obtained saliency is closer to the ground-truth than using separately RF or graph-based saliency computation.

5. Experiments

The performance of the proposed approach is evaluated on five of the most used datasets in the literature and compared to recent methods dealing with SOD. The properties of these datasets are as follows:

(1) *Microsoft MSRA-B* [47]: contains 5000 color images including natural scenes, animals, indoor, outdoor, etc. The ground truth of this dataset is provided in [35].

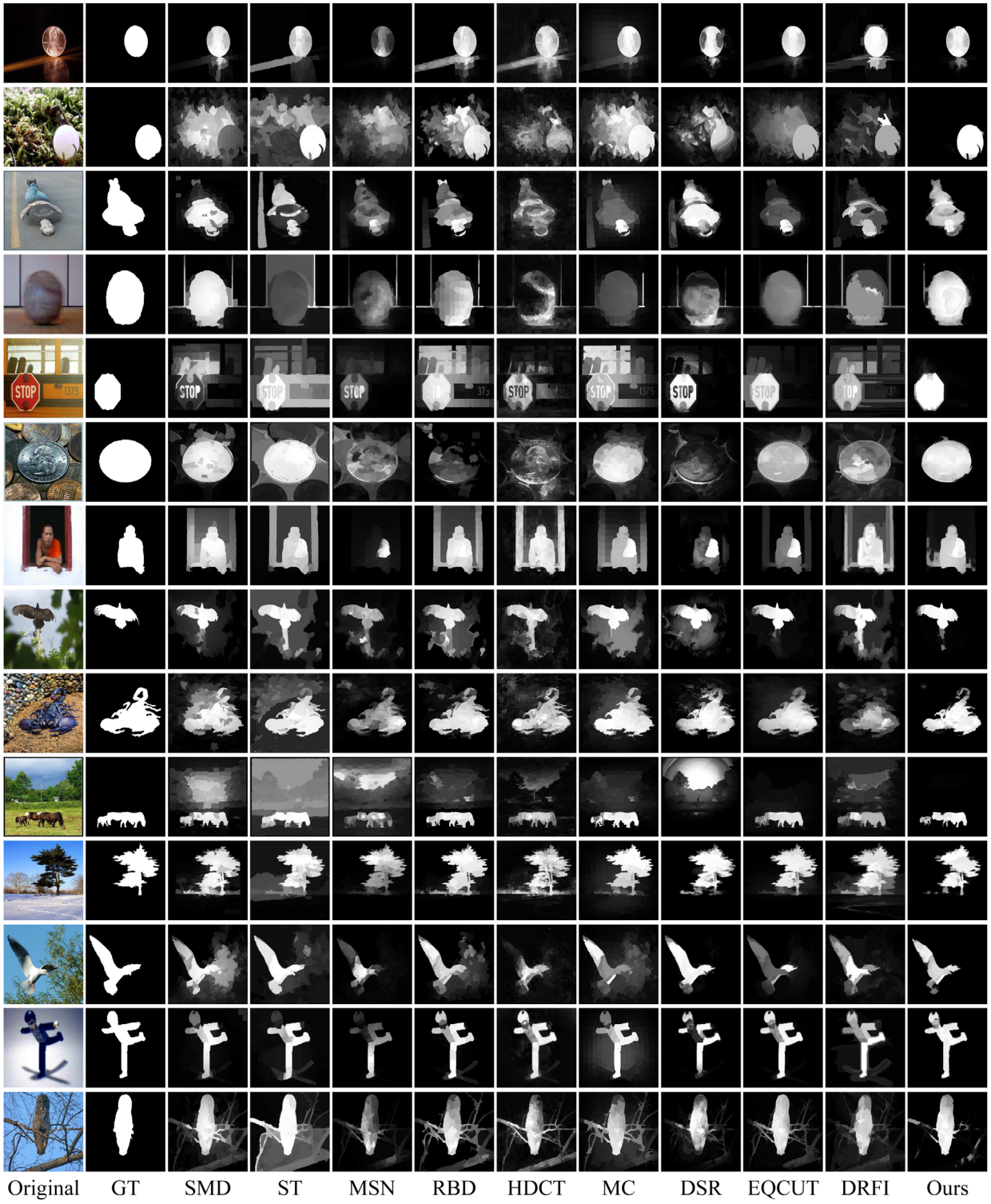

Fig. 24. Visual comparison between our proposed method to best state-of-the-art approaches using the MSRA-B dataset.

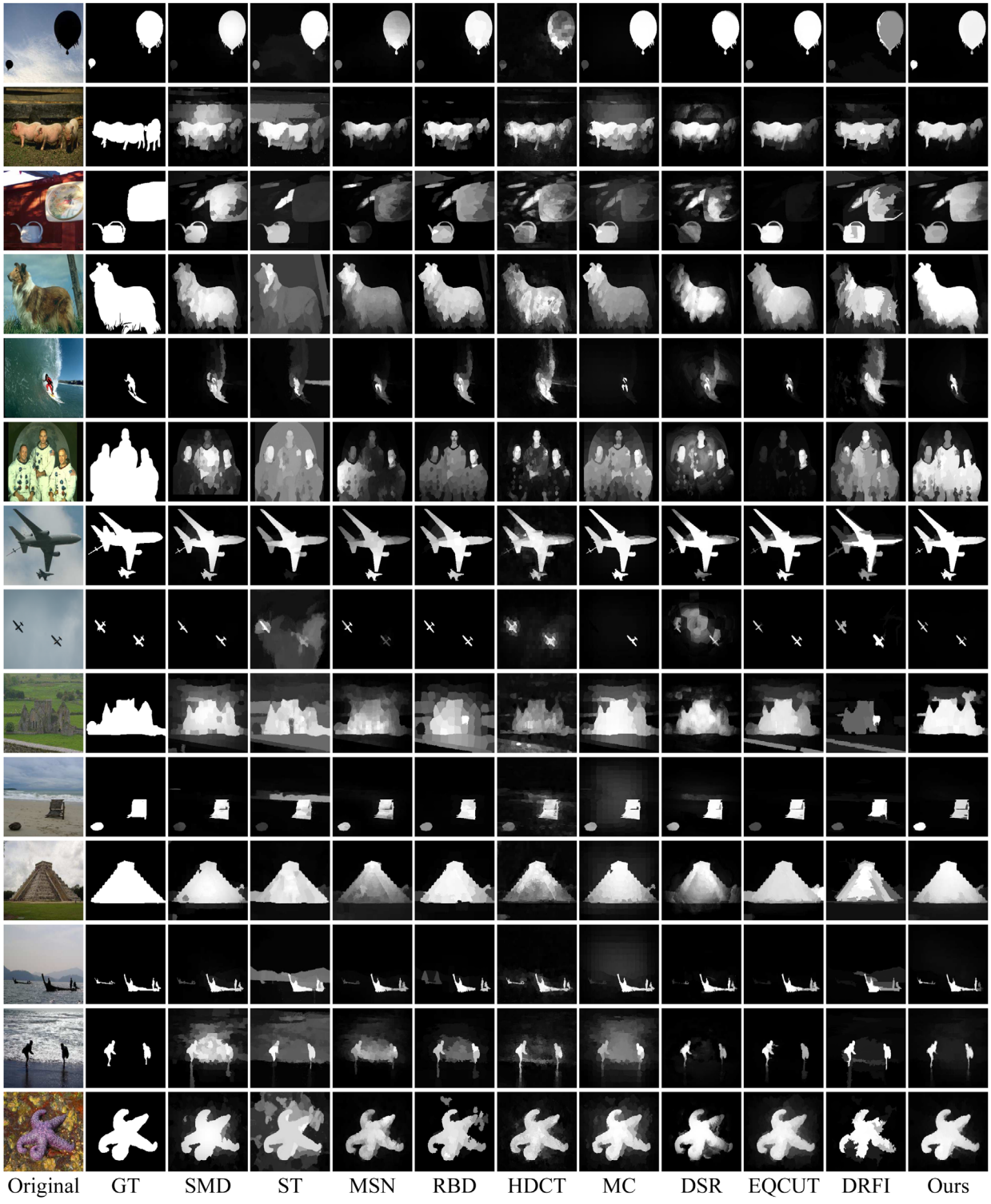

Fig. 25. Visual comparison between our proposed method to best state-of-the-art approaches using SOD, SED1 and SED2 datasets.

(2) *ASD* [2]: contains 1000 images provided by [2] with accurate human-labelled masks for salient objects. It is the most commonly used dataset for evaluation of saliency detection performance and a subset of the MSRA-B dataset.
(3) *SOD* [55]: contains 300 images from the Berkeley segmentation dataset (BSD) [53] for which salient object boundaries are marked by seven users, and a unique binary ground truth is generated for each image by combining the marked boundaries.
(4) *SED*1 and *SED*2 [8]: SED1 is a single object database while two objects exist in each image from SED2. Both datasets contain 100 images which are labeled by several subjects. Pixel-wise ground truth annotations for salient objects in all 200 images are provided

Table 1
Average processing time of each task in our WLMR-RF model.

Steps	Task	Time (s)
1.	Segmentation to superpixels	0.135
2.	Graph construction	1.447
3.	Saliency diffusion	0.666
4.	Window generation	0.154
5.	Feature relevance	0.008
6.	RF training and classification and final saliency map	0.805

Algorithm 1. Multi-scale salient object detection.

Data: Image, number of resolutions L, number of superpixels $n_1, \ldots, n_L$.
Result: Salient object map.
– Multiple resolution segmentation into superpixels using the SLIC algorithm;
– Generate the initial saliency map S using Eqs. (2) and (6);
for $iter = 1$: $iter_{max}$ **do**
– Apply Eqs. (9)–(11) to generate a bounding box W;
– Construct W' by combining W and the top–scored windows generated by [83];
– Calculate feature relevance using Eq. (14) and update Eq. (6);
– Generate the saliency map S_2 using Eqs. (2) and (6)
– Refine saliency using random forests by Eqs. (16) and (17)
– Generate the combined saliency map S_3 using Eq. (18);
end

Note that generally having a large number of resolutions gives better results than having only a few number. However, this comes with additional computational cost since it adds new layers to the graph. For our datasets, we observed that considering resolutions above 800 superpixels does not increase results significantly. To make a compromise between saliency accuracy and computational time, we used 4 layers for each SOD, with scales containing 200, 400, 600 and 800 superpixels, respectively.

5.1. Performance measures

Performance evaluation for SOD can be achieved quantitatively by checking consistency between human-labeled ground-truth and saliency maps generated by computational models. For this, several metrics can be used: the *precision–recall (PR)* curve, the F_μ measure, the *receiver operating characteristic (ROC)* curve, *area under the ROC curve (AUC)* and the *mean absolute error (MAE)*. For a given a saliency map, each pixel has value in the range [0, 255]. We use a threshold T to generate binary mask s assigning 1 to the object and 0 to the background. By varying T from 1 to 254, we obtain 254 binary masks that are compared with the ground truth.

For each binary mask, the precision (P) represents the percentage of salient pixels correctly assigned and the recall (R) corresponds to the ratio of correctly detected salient pixels to true salient pixels. F_μ is a weighted harmonic mean of P and R computed as follows: $F_\mu = \frac{(1+\mu)\times P\times R}{\mu\times P+R}$ where μ is a positive parameter controlling the importance of P over R (we use $\mu^2 = 0.3$ as suggested by Achanta et al. [2]). The *PR* curve plots the 254 average precision values versus the 254 average recall values. Similarly, the ROC curve plots the 254 average true positive rate (TPR) against the 254 average false positive rate (FPR) which are obtained when we calculate the PR curve. MAE is defined as the average absolute difference between the obtained saliency map and the ground truth. We note that the upper-right corner of the PR graph corresponds to the best theoretical result represented by the coordinates (1, 1). The closer a curve is to this point, the best is the performance of a method. Similarly, the closer a curve is to the upper-left corner of the ROC curve, the best is the performance of a method. To visualize the mean of best F_μ, AUC and MAE values, we use a bar diagram.

5.2. Evaluation of the proposed algorithm

We first construct a *PR* and *ROC* curves as well as bar diagrams for F_μ, *MAE* and *AUC* to compare single layer-graph ranking (SLR) [77] and five versions of our algorithm: (1) *WSLR*: the saliency computed after window narrowing and applying function (1), (2) *MLR*: multi-layer graph based saliency computed using function (2), (3) *WMLR*: multi-layer graph based saliency computed with spatial, FR and boundary refinements, (4) *RF*: random forest based saliency and (5) *WMLR-RF*: multi-layer graph based saliency computed after all refinements steps. We use MSRA-B and ASD datasets to evaluate these methods. The results are shown in Figs. 11 and 12. We can observe that MLR outperforms SLR and WSLR for both datasets while WMLR and WSLR improve considerably the results of MLR and SLR, respectively. Also, applying RF alone does not bring a huge performance. However, the combined WMLR-RF yielded the best performance with best (F_μ, AUC, MAE) values (86.78%, 86.40%, 10.76%) in MSRA-B and (93.93%, 88.96%, 6.67%) in ASD). The second performance is obtained by WMLR with (86.06%,85.82%,11.74%) values for MSRA-B and (93.43%,88.64%,7.74%) for ASD, respectively.

5.3. Comparative evaluation using the MSRA-B and ASD datasets

We compared our algorithm to 22 state-of-the-art methods dealing with SOD: Context and shape prior (CS) [34], Human visual sensitivity and amplitude spectrum (VSAS) [22], High-dimensional color transform and local spatial support (HDCT) [39], Saliency optimization from robust Background detection (RBD) [82], Structured matrix decomposition (SMD) [57], Segmentation driven low-rank matrix recovery (SLMR) [84],Hierarchical saliency detection (HSL) [76], Joint spatial-color constraint and multi-scale segmentation (SCC) [75], Saliency Tree (ST) [49], Unified approach to salient object detection via Low rank matrix recovery (LMR) [65], Bayesian Saliency via Low and Mid Level Cues (BS) [74], single-Layer Graph-Based Manifold ranking (SLR) [77], PCA based method (PCA) [52], Combining simple priors method (CSP) [79], Low-level features based on wavelet transform (WT) [32], multi-size superpixel approach based on multivariate normal distribution estimation (MSN) [81], Salient region detection via simple

Table 2
Average processing time of our method in comparison with some competing methods.

Method	OURS	HDCT	RBD	DRFI	DSR	MC	EQCUT	SLR	SMD
Time (s)	2.74	5.53	0.27	19.34	6.14	0.11	0.852	1.373	1.217
Code	Matlab	Matlab	Matlab	Matlab	Matlab	Matlab	Matlab	Matlab	Matlab + C++

local and global contrast (LGC) [50], Salient Region Detection with Soft Image Abstraction (SIA) [17], Saliency Detection Via Dense and Sparse Reconstruction (DSR) [73], Saliency detection via absorb Markov chain (MC) [35], Extended Quantum Cuts (EQCUT) [9] and a Discriminative Regional Feature Integration Approach (DRFI) [36]. For all methods, we used source codes with default parameters provided by the authors. Note that DRFI is the only method which is based on a training step. The authors use 2500 images from the MSRA-B dataset to train their model.

We can see in Figs. 13, 16 and 15 that for both tested datasets, the *precision–recall* curve generated by WMLR-RF is the highest one and closest to the point (1,1) than for curves obtained by the other methods. For each dataset, the values of F_μ, P and R are shown in Fig. 14 for MSRA-B and in Fig. 17 for ASD, where the highest F_μ are clearly returned by our method. This demonstrates the proficiency of our method for SOD. Note also that even by using MLR, it yielded better performance than several other methods. After our approach, the best results are returned by DRFI, EQCUT, SMD, ST, MSN, HDCT, RBD, MR, HSL, MC, SLR and DSR, respectively.

We then operate a second comparative evaluation of our WMLR-RF and MLR with these 12 methods using the ROC curve and the AUC and MAE measures, respectively. Figs. 18–20 show the obtained performance for each method, respectively. We can note that the ROC curve generated by WMLR-RF is the highest for both MSRA-B and ASD datasets. The highest value of AUC and the lowest value of MAE are also returned by WMLR-RF. We can therefore conclude that our model has achieved the highest F_μ and AUC values and the lowest MAE value on these datasets, which demonstrates the quality of saliency maps.

5.4. Comparative evaluation using the SOD, SED1 and SED2 datasets

To extend our experimental evaluation to the datasets, SOD, SED1 and SED2, we perform a comparative study between our method WMLR-RF and the best comparative ones obtained in Section 5.3, namely DRFI, EQCUT, SMD, ST, MSN, HDCT, RBD, MR, HSL, MC, SLR and DSR. Figs. 21–23 show the obtained results for the three datasets. We can see from that the obtained performance for the twelve methods varies according to the datasets. Our PR curves (part (a) of each figure) as well as our ROC curve (part (c) of each figure) are higher than the other methods, especially in the ranges that are closest to the optimal points ((1, 1) for PR curve and (0, 1) for ROC curve). The bar diagrams show clearly that in all the datasets, our method outperforms the others in terms F_μ, AUC and MAE (see parts (b) and (d) of each figure). In the SOD dataset, after our method, DRFI shows better performance, followed by SMD, EQUT and DRFI in terms of F_μ. Concerning AUC and MAE values, the best methods after ours are DRFI, HDCT and ST, respectively.

For the SED1 dataset, the best F_μ values after our method are those of SMD, EQCUT and DRFI. Concerning AUC and MAE values, the best methods after ours are DRFI, SMD, MC and DHCT. For the SED2 dataset, the best F_μ values after our method are of EQCUT, RBD and DRFI. Concerning AUC and MAE values, the best methods after ours are DRFI, HDCT and ST. Fig. 25 shows some examples of saliency maps generated on the MSRA-B dataset by our method and different state-of-the-art methods. We can observe that our saliency maps have relatively cleaner backgrounds than those generated by the other methods. In addition, our model highlights more accurately the salient objects with well-defined boundaries. Fig. 24 illustrates several examples of saliency maps generated on SOD, SED1 and SED2 datasets. We can observe that our model highlights better salient objects for both large and small scales (see rows 1, 5, 10 and 12, for example). Note also that, contrary to our method, single-graph-based saliency detection method (SLR) and other methods miss small objects in images such as in rows 1, 8, 10, 12 and 13. The same remark holds for examples in rows 2, 6 and 13 in Fig. 25 and row 10 in Fig. 24.

6. Computational time analysis

Table 1 summarizes the average computational time of each task in our WLMR-RF in the MSRA dataset on which each image has generally a resolution of 400×300 or 300×400. We use a computer with an Intel Core I7 2.93 GHZ CPU. We run our method on 64-bits Matlab environment. Table 2 summarizes the average running time of some state-of-the-art methods compared to our method. We can note that the computational time of the proposed method is comparable to that of other methods.

For algorithmic complexity analysis, let n_1, $n_2 \ldots n_L$ be the number of superpixels in resolutions Ω_1,... and Ω_L, respectively. Suppose N_1 is the total number of superpixels and N_2 is the number of pixels in the image. The algorithmic complexity of the SLIC algorithm is linear in the number of pixels. Since we can perform the L segmentations in parallel, the total complexity is $\sim O(N_2)$ for this step.

The graph construction requires the constitution of the matrix of weights which has size $N_1 \times N_1$. For that, a single sweep of the image is first performed to construct the adjacency matrix of the graph. The weights are then calculated using Eq. (6) for only adjacent superpixels. This step is relatively fast since only adjacent graph nodes are considered for weight computation. Therefore, the computational complexity is approximately linear with an upper bound $\sim O((N1(N_1 - 1))/2)$ as the worst case where a node is related to all the others. Note also that weight calculation can be performed in parallel between the different nodes and resolutions.

The saliency diffusion using function (2) requires matrix inversion. As the latter is symmetric and sparse, the computation of the inverse is relatively fast ($\approx$0.84 s in the conventional way). To reduce this time, we use the Cholesky decomposition to bring the computation to $\approx$0.346 s. The complexity of such operation is then reduced from $\sim O(N_1^{2.8})$ to near $\sim O(N_1)$ using parallel computation proposed in [64]. We note that both of the feature relevance and window generation have a linear complexity $\sim O(N_2)$.

Finally, the training of RF classifier has the complexity of $\sim O(N_2 \log N_2)$ (as all the trees are computed in parallel). The complexity of mixture of Gaussian estimation is in the order $\sim O((d + K)n)$ with d being the dimension of the data, K the number of mixture components and n the number of data. We note that this operation is also performed for each border of the image in a parallel way, and all together can be run in parallel with RF classification.

7. Conclusion

We have proposed a method for salient object detection combining multi-layer graph ranking and local–global saliency refinement. Our method is able to localize accurately coarse to fine parts of salient objects and their boundaries. This is achieved thanks to a local–global refinement process using random forests and an objectness-like approach for locating the salient object. We also use a feature weighting scheme and boundary information to obtain clearer object boundaries. Our method allows an effective highlighting of the salient object with well defined boundaries. It allows also detecting large- and tiny-scale salient objects in complex scenes. Experiments on five known datasets have demonstrated our model performance compared to recent state-of-the art methods. Our model can be exploited in several applications such as object segmentation and selective object recognition.

Acknowledgement

This work has been achieved thanks to the support of the Government of Algeria and the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] R. Achanta, F. Estrada, P. Wils, S. Susstrunk, Salient region detection and segmentation, in: International Conference on Computer Vision Systems, 2008, pp. 66–75.
[2] R. Achanta, S. Hemami, F. Estrada, S. Susstrunk, Frequency-tuned salient region detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1597–1604.
[3] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Sösstrunk, SLIC superpixels compared to state-of-the-art superpixel methods, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2274–2282.
[4] B. Alexe, T. Deselaers, V. Ferrari, Measuring the objectness of image windows, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2189–2202.
[5] M.S. Allili, D. Ziou, Object of interest segmentation and tracking by using feature selection and active contours, in: IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
[6] M.S. Allili, D. Ziou, Active contours for video object tracking using region, boundary and shape information, Signal Image Video Process. 1 (2) (2007) 101–117.
[7] M.S. Allili, Wavelet modeling using finite mixtures of generalized Gaussian distributions: application to texture discrimination and retrieval, IEEE Trans. Image Process. 21 (4) (2012) 1452–1464, 12.
[8] S. Alpert, M. Galun, A. Brandt, R. Basri, Image segmentation by probabilistic bottom-up aggregation and cue integration, IEEE Trans. Pattern Anal. Mach. Intell. 34 (2) (2012) 315–327.
[9] C. Aytekin, E. Ozan, S. Kiranyaz, M. Gabbouj, Visual saliency by extended quantum cuts, in: IEEE International Conference on Image Processing, 2014, pp. 112–117.
[10] A. Borji, L. Itti, Exploiting local and global patch rarities for saliency detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 478–485.
[11] A. Borji, L. Itti, State-of-the-art in visual attention modeling, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 185–207.
[12] A. Borji, M.-M. Cheng, H. Jiang, J. Li, Salient Object Detection: A Survey. [online] Available: arXiv:1411.5878 2014.
[13] A. Borji, M.-M. Cheng, H. Jiang, J. Li, Salient object detection: a benchmark, IEEE Trans. Image Process. 24 (12) (2015) 5706–5722.
[14] A. Borji, What is a salient object? A dataset and a baseline model for salient object detection, IEEE Trans. Image Process. 24 (2) (2015) 742–756.
[15] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[16] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, S.-M. Hu, Sketch2Photo: internet image montage, ACM Trans. Graph. 28 (5) (2009), Article 124.
[17] M. Cheng, J. Warrell, S. Zheng, V. Vineet, W. Lin, Efficient salient region detection with soft image abstraction, in: IEEE International Conference on Computer Vision, 2013, pp. 1529–1536.
[18] M.-M. Cheng, N.J. Mitra, X. Huang, P.H.S. Torr, S.-M. Hu, Global contrast based salient region detection, IEEE Trans. Pattern Anal. Mach. Intell. 37 (3) (2015) 569–582.
[19] R.T. Collins, Y. Liu, M. Leordeanu, Online selection of discriminative tracking features, IEEE Trans. Pattern Anal. Mach. Intell. 27 (10) (2005) 1631–1643.
[20] P. Dollár, C.L. Zitnick, Structured forests for fast edge detection, in: IEEE International Conference on Computer Vision, 2013, pp. 1841–1848.
[21] Q. Fan, C. Qi, Two-stage salient region detection by exploiting multiple priors, J. Vis. Commun. Image Represent. 25 (8) (2014) 1823–1834.
[22] Y. Fang, W. Lin, B. Lee, C. Lau, Z. Chen, C. Lin, Bottom-up saliency detection model based on human visual sensitivity and amplitude spectrum, IEEE Trans. Multimed. 14 (1) (2012) 187–198.
[23] J. Feng, Y. Wei, L. Tao, C. Zhang, J. Sun, Salient object detection by composition, in: IEEE International Conference on Computer Vision, 2011, pp. 1028–1035.
[24] H. Fu, Z. Chi, D. Feng, Attention-driven image interpretation with application to image retrieval, Pattern Recognit. 39 (9) (2006) 1604–1621.
[25] K. Fu, Fu, C. Gong, J. Yang, Y. Zhou, I.Y. Gu, Superpixel-based color contrast and color distribution driven salient object detection, Signal Process.: Image Commun. 28 (10) (2013) 1448–1463.
[26] S. Goferman, L. Zelnik-Manor, A. Tal, Context-aware saliency detection, IEEE Trans. Pattern Anal. Mach. Intell. 34 (10) (2011) 1915–1926.
[27] V. Gopalakrishnan, Y. Hu, D. Rajan, Salient region detection by modeling distributions of color and orientation, IEEE Trans. Multimed. 11 (5) (2009) 892–905.
[28] 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.
[29] J. Harel, C. Koch, P. Perona, Graph-based visual saliency, in: Advances in Neural Information Processing Systems, vol. 19, no. 12, 2007, pp. 545–552.
[30] X. Hou, L. Zhang, Saliency detection: a spectral residual approach, in: IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
[31] F. Huazhu, C. Xiaochun, T. Zhuowen, Cluster-based co-saliency detection, IEEE Trans. Image Process. 22 (10) (2013) 3766–3778.
[32] N. Imamoglu, W. Lin, Y. Fang, A saliency detection model using low-level features based on wavelet transform, IEEE Trans. Multimed. 15 (1) (2013) 96–105.
[33] L. Itti, C. Koch, E. Niebur, A model of saliency based visual attention for rapid scene analysis, IEEE Trans. Pattern Anal. Mach. Intell. 20 (11) (1998) 1254–1259.
[34] H. Jiang, J. Wang, Z. Yuan, T Liu, N. Zheng, Automatic salient object segmentation based on context and shape prior, in: British Machine Vision Conference, 2011, pp. 1–12.
[35] B. Jiang, L. Zhang, H. Lu, C. Yang, M.-H. Yang, Saliency detection via absorbing Markov chain, in: IEEE International Conference on Computer Vision, 2013, pp. 1665–1672.
[36] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, S. Li, Salient object detection: a discriminative regional feature integration approach, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2083–2090.
[37] R. Ju, Y. Liu, T. Ren, L. Ge, G. Wu, Depth-aware salient object detection using anisotropic center-surround difference, Signal Process.: Image Commun. 38 (2015) 115–126.
[38] T. Kailath, The divergence and Bhattacharyya distance measures in signal selection, IEEE Trans. Commun. Technol. 15 (1) (1967) 52–60.
[39] J. Kim, D. Han, Y.W. Tai, J. Kim, Salient region detection via high-dimensional color transform and local spatial support, IEEE Trans. Image Process. 25 (1) (2016) 9–23.
[40] D.A. Klein, S. Frintrop, Center-surround divergence of feature statistics for salient object detection, in: IEEE International Conference on Computer Vision, 2011, pp. 2214–2219.
[41] G. Larivière, M.S. Allili, A learning probabilistic approach for object segmentation, in: IEEE Canadian Conference on Computer and Robot Vision, 2012, pp. 86–93.
[42] Y. Li, Y. Zhou, J. Yan, Z. Niu, J. Yang, Visual saliency based on conditional entropy, in: Asian Conference on Computer Vision, 2010, pp. 246–257.
[43] Y. Li, X. Hou, C. Koch, J-M. Rehg, A-L. Yuille, The secrets of salient object segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 280–287.
[44] J. Li, Y. Tian, T. Huang, Visual saliency with statistical priors, Int. J. Comput. Vis. 107 (3) (2014) 239–253.
[45] J. Li, X. Qian, K. Lan, P. Qi, A. Sharma, Improved image GPS location estimation by mining salient features, Signal Process.: Image Commun. 38 (2015) 141–150.
[46] T. Liu, J. Sun, N. Zheng, X. Tang, H.Y. Shum, Learning to detect a salient object, in: IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
[47] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, H. Shum, Learning to detect a salient object, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2) (2011) 353–367.
[48] P. Liu, M. Reale, X. Zhang, L. Yin, Saliency-guided 3D head pose estimation on 3D expression models, in: ACM International Conference on Multimodal Interaction, 2013, pp. 75–78.
[49] Z. Liu, W. Zou, O. Le Meur, Saliency tree: a novel saliency detection framework, IEEE Trans. Image Process. 23 (5) (2014) 1937–1952.
[50] J. Liu, S. Wang, Salient region detection via simple local and global contrast representation, Neurocomputing 147 (5) (2015) 435–443.
[51] Y.-F. Ma, X.-S. Hua, L. Lu, H.-J. Zhang, A generic framework of user attention model and its application in video summarization, IEEE Trans. Multimed. 7 (5) (2005) 907–919.
[52] R. Margolin, A. Tal, L.Z. Manor, What makes a patch distinct? in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1139–1146.
[53] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: IEEE International Conference on Computer Vision, 2001, pp. 416–423.
[54] G. McLachlan, D. Peel, Finite Mixture Models, John Wiley and Sons, New York, 2000.
[55] V. Movahedi, J.H. Elder, Design and perceptual validation of performance measures for salient object segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2010, pp. 49–56.
[56] T. Ojala, M. Pietikäinen, T. Mäenpää, Gray scale and rotation invariant texture classification with local binary patterns, in: European Conference on Computer Vision, 2000, pp. 404–420.
[57] H. Peng, B. Li, R. Ji, W. Hu, W. Xiong, C. Lang, Salient object detection via low-rank and structured sparse matrix decomposition, in: AAAI Conference on Artificial Intelligence, 2013, pp. 796–802.
[58] B. Pepik, M. Stark, P. Gehler, B. Schiele, Teaching 3D geometry to deformable part models, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3362–3369.
[59] F. Perazzi, P. Krähenbül, Y. Pritch, A. Hornung, Saliency filters: contrast based filtering for salient region detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 733–740.
[60] X. Qian, J. Han, G. Cheng, L. Guo, Optimal contrast based saliency detection, Pattern Recognit. Lett. 34 (11) (2013) 1270–1278.
[61] E. Rahtu, J. Kannala, M. Salo, J. Heikkil, Segmenting salient objects from images and videos, in: European Conference on Computer Vision, 2010, pp. 366–379.
[62] K. Rapantzikosa, N. Tsapatsoulisb, Y. Avrithisa, S. Kolliasa, Spatiotemporal saliency for video classification, Signal Process.: Image Commun. 24 (7) (2009) 557–571.
[63] H.J. Seo, P. Milanfar, Static and space–time visual saliency detection by self-resemblance, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2009, pp. 45–52.

[64] G. Sharma, A. Agarwala, B. Bhattacharya, A fast parallel Gauss Jordan algorithm for matrix inversion using CUDA, Comput. Struct. 128 (2013) 31–37.
[65] X. Shen, Y. Wu, A unified approach to salient object detection via low rank matrix recovery, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 853–860.
[66] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, A. Blake, Real-time human pose recognition in parts from single depth images, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 1297–1304.
[67] V. Vilaplana, Saliency maps on image hierarchies, Signal Process.: Image Commun. 38 (2015) 84–99.
[68] D. Walther, L. Itti, M. Riesenhuber, T. Poggio, C. Koch, Attentional selection for object recognition—a gentle way, in: Biologically Motivated Computer Vision, Lecture Notes in Computer Science, vol. 2525, 2002, pp. 472–479.
[69] X. Wang, X. Bai, W. Liu, L.J. Latecki, Feature context for image classification and object detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 961–968.
[70] M. Wang, J. Konrad, P. Ishwar, K. Jing, H.A. Rowley, Image saliency: from intrinsic to extrinsic context, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011 pp. 417–424.
[71] P. Wang, J. Wang, G. Zeng, J. Feng, H. Zha, S. Li, Salient object detection for searched web images via global saliency, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3194–3201.
[72] Y. Wei, F. Wen, W. Zhu, J. Sun, Geodesic saliency using background priors, in: European Conference on Computer Vision, 2012, pp. 29–42.
[73] L. Xiaohui, L. Huchuan, Y. Ming-Hsuan, Z. Lihe, R. Xiang, Saliency detection via dense and sparse reconstruction, in: International Conference on Computer Vision (ICCV), 2013.
[74] Y. Xie, H. Lu, M. Yang, Bayesian saliency via low and mid level cues, IEEE Trans. Image Process. 22 (5) (2012) 1689–1698.
[75] L. Xu, H. Li, L. Zeng, K.N. Ngan, Saliency detection using joint spatial-color constraint and multi-scale segmentation, Visual Commun. Image Represent. (2013) 465–476.
[76] Q. Yan, L. Xu, J. Shi, J. Jia, Hierarchical saliency detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1155–1162.
[77] C. Yang, L. Zhang, H. Lu, X. Ruan, Saliency detection via graph-based manifold ranking, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 3166–3173.
[78] Y. Zhai, M. Shah, Visual attention detection in video sequences using spatiotemporal cues, in: ACM International Conference on Multimedia, 2006, pp. 815–824.
[79] L. Zhang, Z. Gu, H. Li, SDSP: a novel saliency detection method by combining simple priors, in: IEEE International Conference on Image Processing, 2013, pp. 171–175.
[80] D. Zhou, O. Bousquet, T.N. Lal, J. Weston, B. Schölkopf, Learning with local and global consistency, in: Neural Information Processing Systems, 2004, pp. 321–328.
[81] L. Zhu, D.A. Klein, S. Frintrop, Z. Cao, A.B. Cremers, A multi-size superpixel approach for salient object detection based on multivariate normal distribution estimation, IEEE Trans. Image Process. 23 (12) (2014) 5094–5107.
[82] W. Zhu, S. Liang, Y. Wei, J. Sun, Saliency optimization from robust background detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2814–2821.
[83] C.L. Zitnick, P. Dollár, Edge boxes: locating object proposals from edges, in: European Conference on Computer Vision, 2014, pp. 391–405.
[84] W. Zou, K. Kpalma, Z. Liu, J. Ronsin, Segmentation driven low-rank matrix recovery for saliency detection, in: British Machine Vision Conference, 2013, pp. 1–13.