Saliency Detection via Dense and Sparse Reconstruction

Xiaohui Li[1], Huchuan Lu[1], Lihe Zhang[1], Xiang Ruan[2], and Ming-Hsuan Yang[3]
[1]Dalian University of Technology [2]OMRON Corporation [3]University of California at Merced

Abstract

In this paper, we propose a visual saliency detection algorithm from the perspective of reconstruction errors. The image boundaries are first extracted via superpixels as likely cues for background templates, from which dense and sparse appearance models are constructed. For each image region, we first compute dense and sparse reconstruction errors. Second, the reconstruction errors are propagated based on the contexts obtained from K-means clustering. Third, pixel-level saliency is computed by an integration of multi-scale reconstruction errors and refined by an object-biased Gaussian model. We apply the Bayes formula to integrate saliency measures based on dense and sparse reconstruction errors. Experimental results show that the proposed algorithm performs favorably against seventeen state-of-the-art methods in terms of precision and recall. In addition, the proposed algorithm is demonstrated to be more effective in highlighting salient objects uniformly and robust to background noise.

1. Introduction

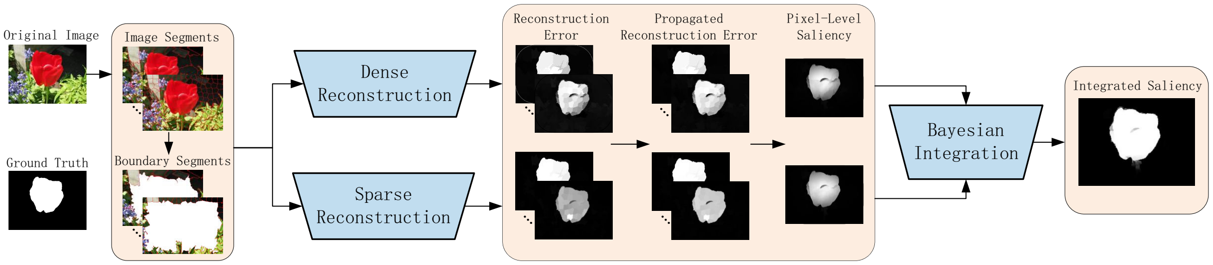
Visual saliency is concerned with the distinct perceptual quality of biological systems which makes certain regions of a scene stand out from their neighbors and catch immediate attention. Numerous biologically plausible models have been developed to explain the cognitive process of humans and animals [12]. In computer vision, more emphasis is paid to detect salient objects in images based on features with generative and discriminative algorithms. Efficient saliency detection plays an important preprocessing role in many computer vision tasks, including segmentation, detection, recognition and compression, to name a few.

Motivated by the neuronal architecture of the early primate vision system, Itti *et al.* [13] define visual attention as the local center-surround difference and propose a saliency model based on multi-scale image features. Rahtu *et al.* [18] propose a saliency detection algorithm by measuring the center-surround contrast of a sliding window over the entire image. While center-surround contrast-based measures are able to detect salient objects, existing bottom-up approaches are less effective in suppressing background pixels. Different from the center-surround contrast, local contrast is measured by comparing a region only with its relevant contexts (defined as a set of region neighbors in the spatial or feature space) [9, 14, 4].

Despite local contrast accords with the neuroscience principle that neurons in the retina are sensitive to regions which locally stand out from their surroundings, global contrast should also be taken into account when one region is similar to its surrounds but still distinct in the whole scene. In other words, global contrast aims to capture the holistic rarity from an image. Recent methods [7, 8] measure global contrast-based saliency based on spatially weighted feature dissimilarities. Perazzi *et al.* [17] formulate saliency estimation using two Gaussian filters by which color and position are respectively exploited to measure region uniqueness and distribution. In [4], global saliency is computed inverse proportionally to the probability of a patch appearing in the entire scene. However, global contrast has its inherent drawbacks. When a foreground region is globally compared with the remaining portion of the scene (which inevitably includes the other foreground regions unless the object boundary is known), its contrast with the background is less distinct and the salient object is unlikely to be uniformly highlighted. In addition, priors or heuristics regarding the likely positions of foreground (e.g., near the image center) and background (e.g., near the image boundary) have been shown to be effective in recent methods [5, 21, 23].

In this paper, we exploit image boundaries as the likely background regions from which templates are extracted. Based on the background templates, we reconstruct the entire image by dense and sparse appearance models from which errors are used as indication of saliency. While dense or sparse representations have been separately applied to saliency detection recently [8, 4], these methods are developed for describing generic scenes. In addition, each image patch is represented by the bases learned from a set of natural image patches rather than other ones directly from the scene, which means that the most relevant visual information is not fully extracted for saliency detection. Therefore, these methods do not uniformly detect salient objects

1550-5499/13 $31.00 © 2013 IEEE
DOI 10.1109/ICCV.2013.370



(a) Background Templates (b) Saliency via Reconstruction Error (c) Bayesian Integration

Figure 1. Main steps of the proposed saliency detection algorithm.

or suppress the background in a scene.

To address the above mentioned issues, we make full use of the visual information by using background templates from each individual image and reconstruct image regions with both dense and sparse representations. In this work, the saliency of each image region is measured by the reconstruction errors using background templates. We exploit a context-based propagation mechanism to obtain more uniform reconstruction errors over the image. The saliency of each pixel is then assigned by an integration of multi-scale reconstruction errors followed by an object-biased Gaussian refinement process. In addition, we present a Bayesian integration method to combine saliency maps constructed from dense and sparse reconstruction. The main steps of the proposed algorithm are shown in Figure 1.

The main contributions of this work are as follows:
1. We propose an algorithm to detect salient objects by dense and sparse reconstruction using the background templates for each individual image, which computes more effective bottom-up contrast-based saliency.
2. A context-based propagation mechanism is proposed for region-based saliency detection, which uniformly highlights the salient objects and smooths the region saliency.
3. We present a Bayesian integration method to combine saliency maps, which achieves more favorable results.

2. Background Templates

To better capture structural information, we first generate superpixels using the simple linear iterative clustering (SLIC) algorithm [3] to segment an input image into multiple uniform and compact regions (i.e., segments). As shown in [4], the use of both Lab and RGB color spaces leads to saliency maps with higher accuracy. We use the mean color features and coordinates of pixels to describe each segment by $\mathbf{x} = \{L, a, b, R, G, B, x, y\}$. The entire image is then represented as $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$, where N is the number of segments and D is the feature dimension.

It has been shown that image boundaries are good visual cues for background models which can be exploited for salient object detection [21, 23]. On the other hand, salient objects are likely to appear at the center of a scene [19, 5]. While these assumptions may not always hold, they nevertheless provide useful visual information which can be utilized to detect salient objects. Motivated by these findings, we extract the D-dimensional feature of each boundary segment as $\mathbf{b}$ and construct the background template set as $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_M]$, where M is the number of image boundary segments. Figure 1(a) shows some boundary templates extracted at different scales (where the non-background regions are masked out).

3. Saliency Measure via Reconstruction Error

We use both dense and sparse reconstruction errors to measure the saliency of each region which is represented by a D-dimensional feature. We note that a dense appearance model renders more expressive and generic descriptions of background templates, whereas a sparse appearance model generates unique and compact representations. It is well known that dense appearance models are more sensitive to noise. For cluttered scenes, dense appearance models may be less effective in measuring salient objects via reconstruction errors. On the other hand, solutions (i.e., coefficients) by sparse representation are less stable (e.g., similar regions may have different sparse coefficients), which may lead to discontinuous saliency detection results. In this work, we use both representations to model regions and measure saliency based on reconstruction errors.

The saliency measures via dense and sparse reconstruction errors are computed as shown in Figure 1(b). First, we reconstruct all the image regions based on the background templates and normalize the reconstruction errors to the range of $[0, 1]$. Second, a propagation mechanism is proposed to exploit local contexts obtained from K-means clustering. Third, pixel-level saliency is computed by taking multi-scale reconstruction errors followed by an object-biased Gaussian refinement process.

3.1. Reconstruction Error

Given the background templates, we intuitively consider image saliency detection as an estimation of reconstruction error on the background, with an assumption that there must be a large difference between the reconstruction errors of foreground and background regions using the same bases. For each region, we compute two reconstruction errors by dense and sparse representation, respectively.

3.1.1 Dense Reconstruction Error

A segment with larger reconstruction error based on the background templates is more likely to be the foreground. Based on this concern, the reconstruction error of each region is computed based on the dense appearance model generated from the background templates $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_M]$, $\mathbf{B} \in \mathbb{R}^{D \times M}$ using Principal Component Analysis (PCA).

The eigenvectors from the normalized covariance matrix of $\mathbf{B}$, $\mathbf{U_B} = [\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_{D'}]$, corresponding to the largest D' eigenvalues, are computed to form the PCA bases of the background templates. With the PCA bases $\mathbf{U_B}$, we compute the reconstruction coefficient of segment i ($i \in [1, N]$).

$$\boldsymbol{\beta}_i = \mathbf{U_B}^\top (\mathbf{x}_i - \bar{\mathbf{x}}), \tag{1}$$

and the dense reconstruction error of segment i is

$$\varepsilon_i^d = \| \mathbf{x}_i - (\mathbf{U_B}\boldsymbol{\beta}_i + \bar{\mathbf{x}}) \|_2^2 , \tag{2}$$

where $\bar{\mathbf{x}}$ is the mean feature of $\mathbf{X}$. The saliency measure is proportional to the normalized reconstruction error (within the range of $[0, 1]$). Figure 2(b) shows some saliency detection results via dense reconstruction. Dense representations model data points with a multivariate Gaussian distribution in the feature space, and thus it may be difficult to capture multiple scattered patterns especially when the number of examples is limited. The middle row of Figure 2 shows an example where some background regions have large reconstruction errors (i.e., inaccurate saliency measure).

3.1.2 Sparse Reconstruction Error

We use the set of background templates $\mathbf{B}$ as the bases for sparse representation, and encode the image segment i by

$$\boldsymbol{\alpha}_i = \underset{\boldsymbol{\alpha}_i}{\operatorname{argmin}} \| \mathbf{x}_i - \mathbf{B}\boldsymbol{\alpha}_i \|_2^2 + \lambda \| \boldsymbol{\alpha}_i \|_1, \tag{3}$$

and the sparse reconstruction error is

$$\varepsilon_i^s = \| \mathbf{x}_i - \mathbf{B}\boldsymbol{\alpha}_i \|_2^2 . \tag{4}$$

Since all the background templates are regarded as the basis functions, sparse reconstruction error can better suppress the background compared with dense reconstruction error especially in cluttered images, as shown in the middle row of Figure 2.

Nevertheless, there are some drawbacks in measuring saliency with sparse reconstruction errors. If some foreground segments are collected into the background templates (e.g., when objects appear at the image boundaries), their saliency measures are close to 0 due to low sparse reconstruction errors. In addition, the saliency measures for the other regions are less accurate due to inaccurate inclusion of foreground segments as part of sparse basis functions. On the other hand, the dense appearance model is not affected by this problem. When foreground segments are mistakenly included in the background templates, the extracted principle components from the dense appearance model may be less effective in describing these foreground regions. As shown in the bottom row of Figure 2, when some foreground segments at the image boundary (e.g., torso and arm) are not detected via sparse reconstruction, these regions are still be detected by the dense counterpart.

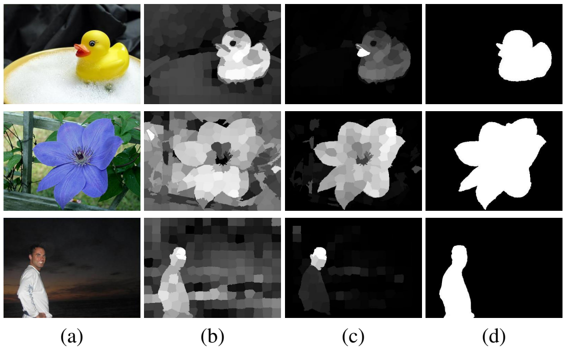


Figure 2. Saliency maps based on dense and sparse reconstruction errors. Brighter pixels indicate higher saliency values. (a) Original images. (b) Saliency maps from dense reconstruction. (c) Saliency maps from sparse reconstruction. (d) Ground truth.

We note sparse reconstruction error is more robust to deal with complicated background, while dense reconstruction error is more accurate to handle the object segments at image boundaries. Therefore, dense and sparse reconstruction errors are complementary in measuring saliency.

3.2. Context-Based Error Propagation

We propose a context-based error propagation method to smooth the reconstruction errors generated by dense and sparse appearance models. Both dense and sparse reconstruction errors of segment i (i.e., ε_i^d and ε_i^s) are denoted by ε_i for conciseness.

We first apply the K-means algorithm to cluster N image segments into K clusters via their D-dimensional features and initialize the propagated reconstruction error of segment i as $\tilde{\varepsilon}_i = \varepsilon_i$. All the segments are sorted in descending order by their reconstruction errors and considered as multiple hypotheses. They are processed sequentially by propagating the reconstruction errors in each cluster. The propagated reconstruction error of segment i belonging to cluster k ($k = 1, 2, ..., K$), is modified by considering its appearance-based context consisting of the other segments in cluster k as follows:

$$\tilde{\varepsilon}_i = \tau \sum_{j=1}^{N_c} \mathrm{w}_{ik_j} \tilde{\varepsilon}_{k_j} + (1 - \tau)\, \varepsilon_i, \tag{5}$$

$$\mathrm{w}_{ik_j} = \frac{\exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_{k_j}\|^2}{2{\sigma_\mathbf{x}}^2})\,(1 - \delta\,(k_j - i))}{\sum\limits_{j=1}^{N_c} \exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_{k_j}\|^2}{2{\sigma_\mathbf{x}}^2})}, \tag{6}$$

where $\{k_1, k_2, ..., k_{N_c}\}$ denote the N_c segment labels in cluster k and τ is a weight parameter. The first term on

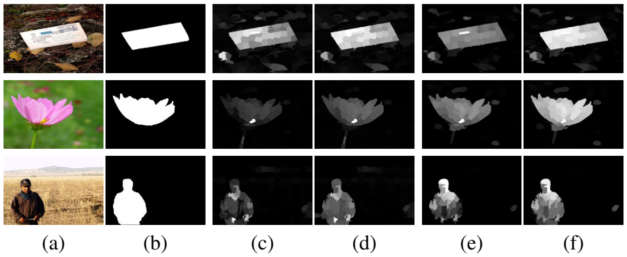


Figure 3. Saliency maps with the context-based error propagation. (a) and (b) are original images and ground truth. (c) and (d) are original and propagated dense reconstruction errors. (e) and (f) are original and propagated sparse reconstruction errors.

the righthand side of Eq. 5 is the weighted averaging reconstruction error of the other segments in the same cluster, and the second term is the initial dense or sparse reconstruction error. That is, for segment i, by considering all the other segments belonging to the same cluster k (i.e., the appearance-based local context), the reconstruction error can be better estimated. The weight of each segment context is defined by its normalized similarity with segment i in Eq. 6, where $\sigma_{\mathbf{x}}^2$ is the sum of the variance in each feature dimension of $\mathbf{X}$ and $\delta(\cdot)$ is the indicator function.

Figure 3 shows three examples where the context-based propagation mechanism smooths the reconstruction errors in a cluster, thereby uniformly highlighting the image objects. The bottom row of Figure 3 presents one case that several segments of the object (e.g., torso) are mistakenly included in the background templates, and therefore they are not correctly identified by the dense and sparse appearance models. Nevertheless, the reconstruction errors of these segments are modified by taking the contributions of their contexts into consideration using Eq. 5.

3.3. Pixel-Level Saliency

For a full-resolution saliency map, we assign saliency to each pixel by integrating results from multi-scale reconstruction errors, followed by refinement with an object-biased Gaussian model.

3.3.1 Multi-Scale Reconstruction Error Integration

To handle the scale problem, we generate superpixels at N_s different scales. We compute and propagate both dense and sparse reconstruction errors for each scale. We integrate multi-scale reconstruction errors and compute the pixel-level reconstruction error by

$$E(z) = \frac{\sum\limits_{s=1}^{N_s} \omega_{zn^{(s)}} \tilde{\varepsilon}_{n^{(s)}}}{\sum\limits_{s=1}^{N_s} \omega_{zn^{(s)}}}, \omega_{zn^{(s)}} = \frac{1}{\left\|\mathbf{f}_z - \mathbf{x}_{n^{(s)}}\right\|_2}, \quad (7)$$

where $\mathbf{f}_z$ is a D-dimensional feature of pixel z and $n^{(s)}$ denotes the label of the segment containing pixel z at scale s. Similarly to [14], we utilize the similarity between pixel z and its corresponding segment $n^{(s)}$ as the weight to average the multi-scale reconstruction errors.

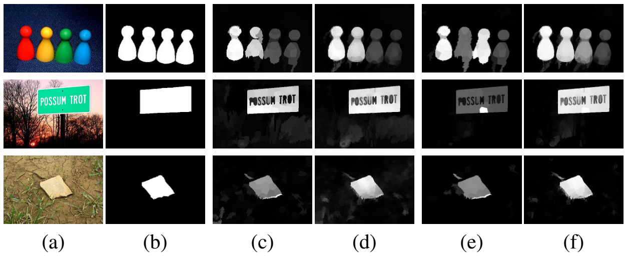


Figure 4. Saliency maps with the multi-scale integration of propagated reconstruction errors. (a) and (b) are original images and ground truth. (c) and (d) are propagated dense reconstruction errors without and with integration. (e) and (f) are propagated sparse reconstruction errors without and with integration.

Figure 4 shows some examples where objects are more precisely identified by the reconstruction errors with multi-scale integration, which suggests the effectiveness of using multi-scale integration mechanism to measure saliency.

3.3.2 Object-Biased Gaussian Refinement

Borji *et al.* show that there is a center bias in some saliency detection datasets [5]. Recently center prior has been used in [8, 14, 19] and usually formulated as a Gaussian model,

$$G(z) = \exp\left[-\left(\frac{(x_z - \mu_x)^2}{2{\sigma_x}^2} + \frac{(y_z - \mu_y)^2}{2{\sigma_y}^2}\right)\right], \quad (8)$$

where $\mu_x = x_c$ and $\mu_y = y_c$ denote the coordinates of the image center and x_z and y_z are the coordinates of pixel z. Since salient objects do not always appear at the image center as Figure 5 shows, the center-biased Gaussian model is not effective and may include background pixels or miss the foreground regions. We use an object-biased Gaussian model G_o with $\mu_x = x_o$ and $\mu_y = y_o$, where x_o and y_o denote the object center derived from the pixel error in Eq. 7:

$$\begin{cases} x_o = \sum\limits_i \frac{E(i)}{\sum\limits_j E(j)} x_i \\ y_o = \sum\limits_i \frac{E(i)}{\sum\limits_j E(j)} y_i \end{cases}. \quad (9)$$

We set $\sigma_x = 0.25 \times H$ and $\sigma_y = 0.25 \times W$, where W and H respectively denote the width and height of an image. With the object-biased Gaussian model, the saliency of pixel z is computed by $S(z) = G_o(z) * E(z)$.

Figure 5 shows an example when the object does not locate at the image center. Comparing the two refined maps of the saliency via dense or sparse reconstruction in the bottom row, the proposed object-biased Gaussian model renders more accurate object center, and therefore better refines the saliency detection results.

4. Bayesian Integration of Saliency Maps

As mentioned in Section 3.1, the saliency measures by dense and sparse reconstruction errors are complementary to each other. To integrate both the saliency measures, we propose an integration method by Bayesian inference.

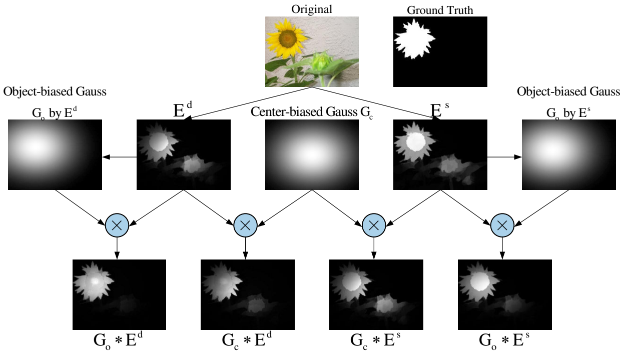


Figure 5. Comparison of center-biased (G_c) and object-biased (G_o) Gaussian refinement. E^d and E^s are the multi-scale integrated dense and sparse reconstruction error maps, respectively.

Recently, the Bayes formula has been used to measure saliency by the posterior probability in [18, 20, 22]:

$$p(F|H(z)) = \frac{p(F)p(H(z)|F)}{p(F)p(H(z)|F) + (1 - p(F))p(H(z)|B)}, \tag{10}$$

where the prior probability $p(F)$ is a uniform [18] or a saliency map [20, 22] and $H(z)$ is a feature vector of pixel z. The likelihood probabilities are computed as:

$$\begin{aligned} p(H(z)|F) &= \prod_{r \in \{L,a,b\}} \frac{N_{b_F(r(z))}}{N_F}, \\ p(H(z)|B) &= \prod_{r \in \{L,a,b\}} \frac{N_{b_B(r(z))}}{N_B}, \end{aligned} \tag{11}$$

where N_F denotes the number of pixels in the foreground and $N_{b_F(r(z))}(r \in \{L, a, b\})$ is the number of pixels whose color features fall into the foreground bin $b_F(r(z))$ which contains feature $r(z)$, while the color distribution histogram of the background is denoted likewise by N_B and $N_{b_B(r(z))}$. In this work, we take one saliency map as the prior and use the other one instead of Lab color information to compute the likelihoods, which integrates more diverse information from different saliency maps.

Given two saliency maps S_1 and S_2 (i.e., from dense and sparse reconstruction), we treat one of them as the prior $S_i (i = \{1, 2\})$ and use the other one $S_j (j \neq i, j = \{1, 2\})$ to compute the likelihood, as shown in Figure 6. First, we threshold the map S_i by its mean saliency value and obtain its foreground and background regions described by F_i and B_i, respectively. In each region, we compute the likelihoods by comparing S_j and S_i in terms of the foreground and background bins at pixel z:

$$p(S_j(z)|F_i) = \frac{N_{b_{F_i}(S_j(z))}}{N_{F_i}}, p(S_j(z)|B_i) = \frac{N_{b_{B_i}(S_j(z))}}{N_{B_i}}. \tag{12}$$

Consequently the posterior probability is computed with S_i as the prior by

$$p(F_i|S_j(z)) = \frac{S_i(z)p(S_j(z)|F_i)}{S_i(z)p(S_j(z)|F_i)+(1-S_i(z))p(S_j(z)|B_i)}. \tag{13}$$

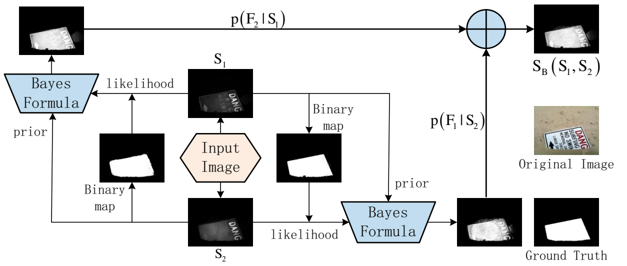


Figure 6. Bayesian integration of saliency maps. The two saliency measures via dense and sparse reconstruction are respectively denoted by S_1 and S_2.

Similarly, the posterior saliency with S_j as the prior is computed. We use these two posterior probabilities to compute an integrated saliency map, $S_B(S_1(z), S_2(z))$, based on Bayesian integration:

$$S_B(S_1(z), S_2(z)) = p(F_1|S_2(z)) + p(F_2|S_1(z)). \tag{14}$$

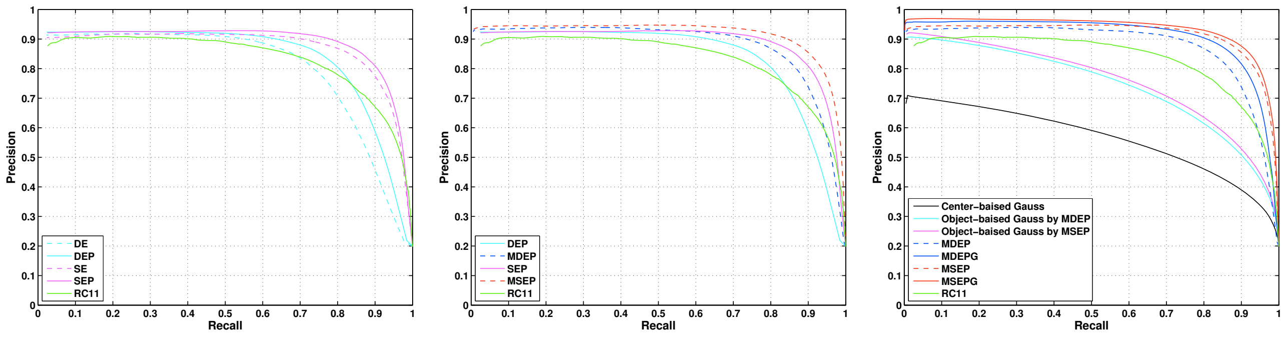
The proposed Bayesian integration of saliency maps is illustrated in Figure 6. It should be noted that Bayesian integration enforces these two maps to serve as the prior and cooperate with each other in an effective manner, which uniformly highlights salient objects in an image.

5. Experiments

We evaluate the proposed algorithm with seventeen state-of-the-art algorithms including IT98 [13], MZ03 [24], LC06 [25], GB06 [10], SR07 [11], AC08 [1], FT09 [2], CA10 [9], RA10 [18], RC11 [7], CB11 [14], SVO11 [6], DW11 [8], SF12 [17], LR12 [19], GS12 [21] and XL13 [22] on three benchmark data sets: ASD, MSRA and SOD.

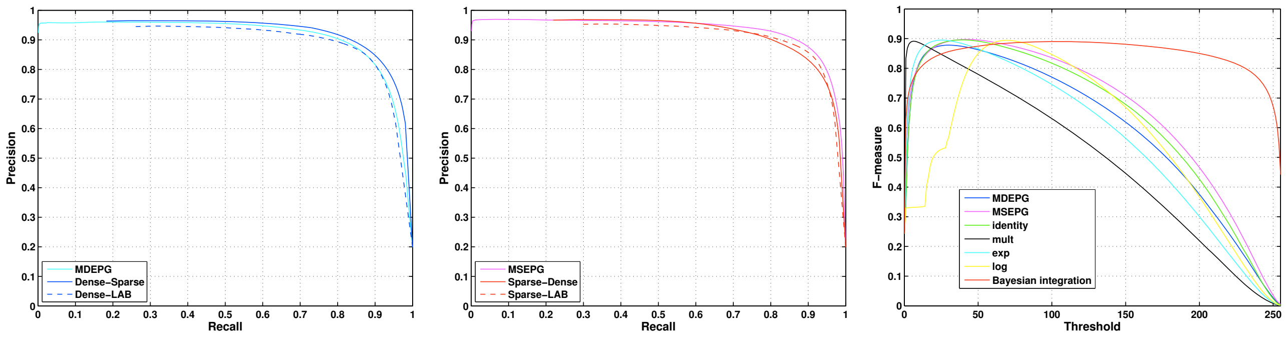
5.1. Data Sets

The MSRA database [15] contains 5000 images, where each one is labeled with bounding boxes by nine subjects. We compute the mean of these bounding boxes as the ground truth data. The ASD database [2] includes 1000 images selected from the MSRA database, where each image is manually segmented into foreground and background. Most images in the MSRA and ASD databases have only one salient object and there are usually strong contrast between objects and backgrounds. In addition, we evaluate the proposed algorithm on the SOD database. The SOD database [16] is based on the Berkeley segmentation dataset where seven subjects are asked to label 300 images with object boundaries. For each object mask from each subject, a consistency score is computed by the other six masks. Similar to [21], the objects with low consistency scores are removed. This dataset is more challenging than the other databases with multiple objects of different sizes and locations in more complicated backgrounds.



(a) (b) (c)

Figure 7. Evaluation of saliency via reconstruction error. (a) Based on the context-based propagation. (b) Based on the multi-scale reconstruction error integration. (c) Based on the object-biased Gaussian refinement. DE: dense reconstruction error; DEP: propagated DE; MDEP: multi-scale integrated DEP; MDEPG: Gaussian refined MDEP. SE: sparse reconstruction error; SEP: propagated SE; MSEP: multi-scale integrated SEP; MSEPG: Gaussian refined MSEP.



(a) (b) (c)

Figure 8. Evaluation of Bayesian integrated saliency. (a) Precision-recall curves of Dense-Sparse and Sparse-Dense Bayesian models. (c) F-measure curves of the proposed Bayesian integrated saliency S_B and four other integrated saliency of MDEPG and MSEPG.

5.2. Implementation Details

Parameter Setting. The two main parameters of our method are the number of clusters K and the weight factor τ in Eq. 5. We set $K = 8$ and $\tau = 0.5$ in all experiments. We vary K (K=$4, 6, 8, 10, 12$) and τ (τ=$0.1, 0.3, 0.5, 0.7, 0.9$) in experiments, and observe that the saliency results are insensitive to either parameter. The parameter λ of Eq. 3, is empirically set to 0.01. For dense reconstruction, we use the eigenvectors corresponding to the biggest eigenvalues which retain 95% of the energy. For multi-scale reconstruction errors, we generate superpixels at eight different scales respectively with 50 to 400 superpixels.

Evaluation Metrics. We evaluate all saliency detection algorithms in terms of precision-recall curve and F-measure. For each method, a binary map is obtained by segmenting each saliency map with a given threshold $T \in [0, 255]$ and then compared with the ground truth mask to compute the precision and recall for an image. The mean precisions and recalls of all images are then depicted in the precision-recall curve. The evaluation of F-measure is similar to [2]. We first use the mean-shift algorithm to segment the original image and extract the mean saliency of each segment. We then obtain the binary map by thresholding the segments using twice the mean saliency value. For each binary map, we compute the precision, recall and F-measure, F_{γ}=$\frac{(1+\gamma^2)\mathrm{Precision}\times\mathrm{Recall}}{\gamma^2\mathrm{Precision}+\mathrm{Recall}}$, where we set $\gamma^2 = 0.3$ to emphasize precision [2]. We compare the mean precision, recall and F-measure with a bar graph. For each image in the MSRA database which is labeled with a bounding box (rather than precise object contour), we fit a rectangle to the thresholded saliency map for evaluation, similar to [5].

5.3. Experimental Results

In addition to comparisons with seventeen state-of-the-art approaches, we also evaluate each individual component of the proposed algorithm on the ASD database.

Saliency Detection via Reconstruction Error. We evaluate the contribution of the context-based propagation, multi-scale reconstruction error integration and object-biased Gaussian refinement respectively in Figure 7. The approach in [7] (referred as RC11) is also presented as a baseline model for comparisons. Figure 7(a) shows that the sparse reconstruction error based on background templates achieves better accuracy in detecting salient objects than RC11 [7], while the dense one is comparable with it. The context-based reconstruction error propagation method uses

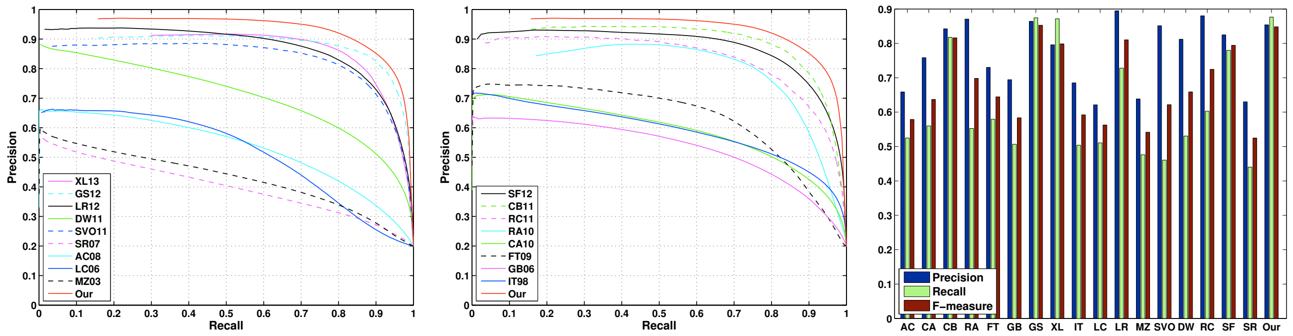
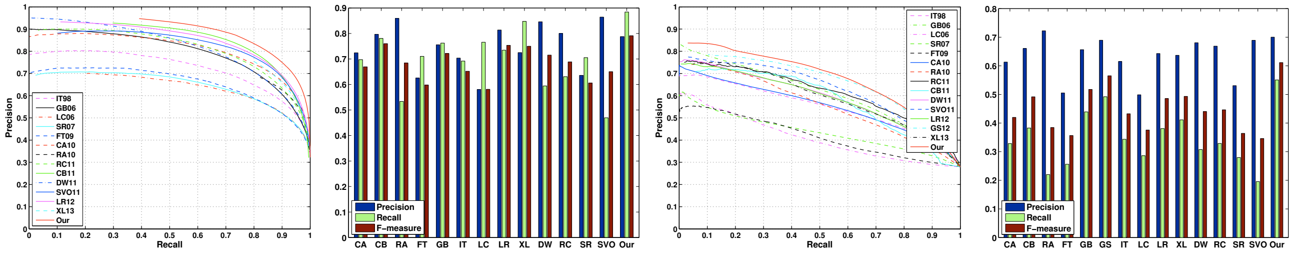


Figure 9. Performance of the proposed method compared with seventeen state-of-the-art methods on the ASD database.



(a) MSRA

(b) SOD

Figure 10. Performance of the proposed algorithm compared with other methods on the MSRA and SOD databases.

segment contexts through K-means clustering to smooth the reconstruction errors and minimize the detection mistakes introduced by the object segments in background templates with improved performance (Figure 7(a)). The reconstruction error of a pixel is assigned by integrating the multi-scale reconstruction errors, which helps generate more accurate and uniform saliency maps. Figure 7(b) shows the improved performance due to the integration of multi-scale reconstruction errors. Figure 7(c) shows that the object-biased Gaussian model further refines the results and performs better than the center-biased one.

Bayesian Integrated Saliency Detection. In Section 4, we discuss that the posterior probability can be more accurate with likelihood computed by a saliency map rather than the CIELab color space on the condition of the same prior in the Bayes formula. We present experimental results in which we treat the saliency map by dense (or sparse) reconstruction as the prior, and use the other saliency map by sparse (or dense) reconstruction and Lab color to compute the likelihood probability, denoted respectively by Dense-Sparse (or Sparse-Dense) and Dense-Lab (or Sparse-Lab) in Figure 8(a) (or (b)). Figure 8(a) shows that with the saliency via dense reconstruction as the prior, the result with the likelihood based on sparse reconstruction (Dense-Sparse) is more accurate than that with the CIELab color space (Dense-Lab). While using the saliency map based on sparse reconstruction as the prior, the result with the likelihood based on dense reconstruction (Sparse-Dense) is comparable to that with the CIELab color space (Sparse-Lab) as shown in Figure 8(b). Although both precisions of the Sparse-Dense and Sparse-Lab models are lower than that of the prior, the recalls are improved, which also suggests the fact that the likelihood probability may introduce noise that has been removed by the prior thus lead to worse posterior than the prior in certain cases.

In addition, we also present the F-measure curve depicted by the mean F-measure at each threshold from 0 to 255 in Figure 8(c). We evaluate the performance of Bayesian integrated saliency map S_B by comparing it with the integration strategies formulated in [5]:

$$S_c = \frac{1}{Z}\sum_i Q\left(S_i\right) \quad \text{or} \quad S_c = \frac{1}{Z}\prod_i Q\left(S_i\right), \tag{15}$$

where Z is the partition function. In Figure 8(c), we denote the linear summation S_c with $Q(x) = \{x, \exp(x), -1/\log(x)\}$ respectively by *identity*, *exp* and *log*, while denote the accumulation S_c with $Q(x) = x$ by *mult*. Figure 8(c) shows that the F-measure of the proposed Bayesian integrated saliency map is higher than the other methods at most thresholds, which demonstrates the effectiveness of Bayesian integration.

Comparisons with State-of-the-Art Methods. We present the evaluation results of the proposed method compared with the state-of-the-art saliency detection methods on the ASD database in Figure 9, and the MSRA and SOD databases in Figure 10. The precision-recall curves show our method achieves consistent and favorable performance against the state-of-the-art methods. In the bar graphs, the precision, recall and F-measure of the proposed algorithm are comparable with those of the other algorithms, especially with higher recall and

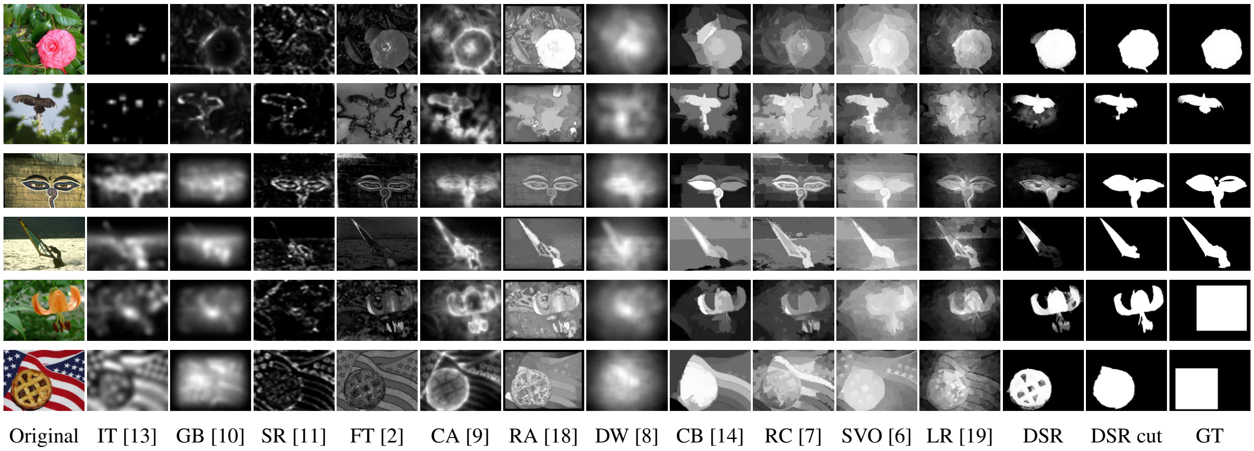


Figure 11. Comparisons of saliency maps. Top, middle and bottom two rows are images from the ASD, SOD and MSRA data sets. DSR: the proposed algorithm based on dense and sparse reconstruction. DSR cut: cut map using the generated saliency map. GT: ground truth.

F-measure. Figure 11 shows that our model generates more accurate saliency maps with uniformly highlighted foreground and well suppressed background. Our Matlab implementation is available at `http://ice.dlut.edu.cn/lu/publications.html` or `http://faculty.ucmerced.edu/mhyang/pubs.html`.

6. Conclusions

In this paper, we present a saliency detection algorithm via dense and sparse reconstruction based on the background templates. A context-based mechanism is designed to propagate the reconstruction errors. The pixel-level saliency is then computed by an integration of multi-scale reconstruction errors followed by an object-biased Gaussian refinement. To combine the two saliency maps via dense and sparse reconstruction, we introduce a Bayesian integration method which performs better than the conventional integration strategy. Experimental results show the performance improvement of the proposed method compared to seventeen state-of-the-art models.

Acknowledgements. X. Li and H. Lu are supported by the Natural Science Foundation of China (NSFC) #61071209 and #61272372. L. Zhang is supported by the NSFC #61371157. M.-H. Yang is supported by the NSF CAREER Grant #1149783 and NSF IIS Grant #1152576.

References

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

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

[3] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels. Technical Report 149300, EPFL, 2010.

[4] A. Borji and L. Itti. Exploiting local and global patch rarities for saliency detection. In *CVPR*, pages 478–485, 2012.

[5] A. Borji, D. N. Sihite, and L. Itti. Salient object detection: A benchmark. In *ECCV*, 2012.

[6] K.-Y. Chang, T.-L. Liu, H.-T. Chen, and S.-H. Lai. Fusing generic objectness and visual saliency for salient object detection. In *ICCV*, pages 914–921, 2011.

[7] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu. Global contrast based salient region detection. In *CVPR*, pages 409–416, 2011.

[8] L. Duan, C. Wu, J. Miao, L. Qing, and Y. Fu. Visual saliency detection by spatially weighted dissimilarity. In *CVPR*, pages 473–480, 2011.

[9] S. Goferman, L. Zelnik-Manor, and A. Tal. Context-aware saliency detection. In *CVPR*, pages 2376–2383, 2010.

[10] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. In *NIPS*, 2006.

[11] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *CVPR*, 2007.

[12] L. Itti and C. Koch. Computational modeling of visual attention. *Nature Reviews Neuroscience*, 2(3):194–201, 2001.

[13] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *PAMI*, 20:1254–1259, 1998.

[14] H. Jiang, J. Wang, Z. Yuan, T. Liu, N. Zheng, and S. Li. Automatic salient object segmentation based on context and shape prior. In *BMVC*, 2011.

[15] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. In *CVPR*, 2007.

[16] V. Movahedi and J. H. Elder. Design and perceptual validation of performance measures for salient object segmentation. In *POCV*, pages 49–56, 2010.

[17] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung. Saliency filters: Contrast based filtering for salient region detection. In *CVPR*, pages 733–740, 2012.

[18] E. Rahtu, J. Kannala, M. Salo, and J. Heikkilä. Segmenting salient objects from images and videos. In *ECCV*, pages 366–379, 2010.

[19] X. Shen and Y. Wu. A unified approach to salient object detection via low rank matrix recovery. In *CVPR*, pages 853–860, 2012.

[20] J. Sun, H. Lu, and S. Li. Saliency detection based on integration of boundary and soft-segmentation. In *ICIP*, pages 1085–1088, 2012.

[21] Y. Wei, F. Wen, W. Zhu, and J. Sun. Geodesic saliency using background priors. In *ECCV*, pages 29–42, 2012.

[22] Y. Xie, H. Lu, and M.-H. Yang. Bayesian saliency via low and mid level cues. *TIP*, 22(5):1689–1698, 2013.

[23] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, 2013.

[24] M. Yifeng and Z. Hejie. Contrast-based image attention analysis by using fuzzy growing. In *ACM Multimedia*, pages 374–381, 2003.

[25] Y. Zhai and M. Shah. Visual attention detection in video sequences using spatiotemporal cues. In *ACM Multimedia*, pages 815–824, 2006.