Co-Bootstrapping Saliency

Huchuan Lu, *Senior Member, IEEE*, Xiaoning Zhang, Jinqing Qi, *Member, IEEE*,
Na Tong, Xiang Ruan, and Ming-Hsuan Yang, *Senior Member, IEEE*

***Abstract*— In this paper, we propose a visual saliency detection algorithm to explore the fusion of various saliency models in a manner of bootstrap learning. First, an original bootstrapping model, which combines both weak and strong saliency models, is constructed. In this model, image priors are exploited to generate an original weak saliency model, which provides training samples for a strong model. Then, a strong classifier is learned based on the samples extracted from the weak model. We use this classifier to classify all the salient and non-salient superpixels in an input image. To further improve the detection performance, multi-scale saliency maps of weak and strong model are integrated, respectively. The final result is the combination of the weak and strong saliency maps. The original model indicates that the overall performance of the proposed algorithm is largely affected by the quality of weak saliency model. Therefore, we propose a co-bootstrapping mechanism, which integrates the advantages of different saliency methods to construct the weak saliency model thus addresses the problem and achieves a better performance. Extensive experiments on benchmark data sets demonstrate that the proposed algorithm outperforms the state-of-the-art methods.**

***Index Terms*— Saliency detection, weak saliency model, strong saliency model, co-bootstrapping.**

I. Introduction

As AN important preprocessing step in computer vision problems, saliency detection has attracted much attention in recent years. The saliency value of a pixel or region is a metric that describes how much it catches one's attention when he or she looks at an image. However, due to many uncertainties of how human beings understand image contents, although significant progress has been made in latest few years, saliency detection remains a challenging task in computer vision.

Recently, many salient object detection methods have been proposed which can be categorized as bottom-up stimuli-driven [1]–[30] and top-down task-driven [31]–[39] methods. Bottom-up methods are usually based on low-level visual information and are more effective in detecting fine details rather than global shape information. In contrast, top-down saliency models are able to detect objects of certain sizes and categories based on more representative features from training samples. However, the detection results from top-down methods tend to be coarse with fewer details. In terms of computational complexity, bottom-up methods are often more efficient than top-down approaches.

In this paper, we propose a novel algorithm for salient object detection via bootstrap learning [40]. To address the problems of noisy detection results and limited representations from bottom-up methods, we present a learning approach to exploit multiple features. However, unlike existing top-down learning-based methods, the proposed algorithm is bootstrapped with samples from a bottom-up model, thereby alleviating the time-consuming off-line training process or labeling positive samples manually.

Our previous work [41] shows that the proposed bootstrap learning algorithm is effective for saliency detection. At the same time, it also demonstrates that the overall performance of the bootstrap learning algorithm hinges on the quality of the weak saliency model. If a weak saliency model does not perform well, the proposed algorithm is likely to fail as an insufficient number of good training samples can be collected for constructing the strong model for a specific image. Therefore, we propose a co-bootstrapping mechanism which explores complementary effects of various saliency methods to address the problem in our previous work. It is observed that there are many proposed saliency detection methods which have both advantages and disadvantages. We integrate different saliency methods in a manner of bootstrap learning, which combines the strengths of various methods and overcomes the problem caused by weak saliency model when it fails to offer enough good training samples. To better integrate advantages of different saliency methods, we propose two bootstrapping strategies: *Co-map bootstrapping* and *Co-sample bootstrapping*. The former keeps the integrity of these methods while the latter mines more potential information.

The results show that the bootstrap learning algorithm performs favorably against the state-of-the-art saliency detection methods. It also demonstrates that existing saliency methods have complementary effects which can be exploited for better detection performance and our co-bootstrapping algorithms provide us with an effective way to combine strengths of existing saliency methods.

There are three main contributions in this work:
1. We propose a bootstrap learning algorithm for salient object detection in which both weak and strong models are exploited.

Manuscript received April 5, 2016; revised August 3, 2016; accepted September 5, 2016. Date of publication November 11, 2016; date of current version November 29, 2016. This work was supported by the National Natural Science Foundation of China under Grant 61528101 and Grant 61472060. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Chang-Su Kim.

H. Lu, X. Zhang, J. Qi, and N. Tong are with the School of Information and Communication Engineering, Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: lhchuan@dlut.edu.cn; jinqing@dlut.edu.cn).

X. Ruan is with Tiwaki Company, Kusatsu, Shiga 525-0035, Japan (e-mail: ruanxiang@tiwaki.com).

M.-H. Yang is with the Department of Electrical Engineering and Computer Science, University of California at Merced, Merced, CA 95344 USA (e-mail: mhyang@ucmerced.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2627804

2. The proposed bootstrap learning algorithm can be easily applied to other existing algorithms to improve their performance.
3. The co-bootstrapping mechanism is proposed to integrate advantages of different saliency methods, which can achieve a better performance.

The paper is organized as follows: In Section II, some previous works related to our paper are introduced. Then, the proposed bootstrap saliency method is presented in Section III. In Section IV, we display and analyze the experimental results. Finally, we conclude the whole paper in section V.

II. Related Works

In recent years, numerous bottom-up saliency detection methods have been proposed. Itti *et al.* [1] propose a saliency model based on a neural network that integrates three feature channels over multiple scales for rapid scene analysis. While it is able to identify salient pixels, the results contain a significant amount of false detections. Saliency models based on Bayesian inference have been proposed in [2], [13], and [23]. In [3], the low-level saliency stimuli and the shape prior are integrated using an iterative energy minimization measure. While the above-mentioned contrast-based methods are simple and effective, pixels within the salient objects are not always highlighted well. In [12], Wei *et al.* focus on the background instead of the foreground and build a saliency detection model based on two background priors, i.e., boundary and connectivity. Cheng *et al.* [16] utilize a soft abstraction method to remove unnecessary image details and produce perceptually accurate salient regions. A graph-based bottom-up method is proposed using manifold ranking [18]. In [29], Qin *et al.* propose a novel saliency model based on cellular automata to intuitively detect the salient object. Recently, Kong *et al.* [30] presents a method that can effectively mine the saliency patterns of initial saliency maps.

Compared to bottom-up approaches, considerable efforts have been made on top-down saliency models. In [31], Zhang *et al.* integrate both the top-down and bottom-up information to construct a Bayesian-based top-down model where saliency is computed locally. A saliency model based on the Conditional Random Field is formulated with latent variables and a discriminative dictionary in [32]. Jiang *et al.* [33] propose a learning-based method by regarding saliency detection as a regression problem where the saliency detection model is constructed based on the integration of numerous descriptors extracted from training samples with ground truth labels. In [38], Wang *et al.* jointly learn a ranker and a distance metric to construct a saliency map by top-ranked region proposals. Recently, some CNN-based deep learning methods are proposed. In [34], two deep neural networks are trained to learn local and global features respectively. Li and Yu [35] introduce a neural network architecture to extract multiscale deep features from which a high-quality visual saliency model can be learned. In [37], a multi-task deep saliency model based on a fully convolutional neural network (FCNN) is proposed to model the semantic properties of salient objects.

Considering these two categories bring forth different properties of efficient and effective salient detection algorithms, we propose a bootstrap learning approach which exploits the strengths of both bottom-up contrast-based saliency models and top-down learning methods. Furthermore, it is observed that the overall performance of the proposed method depends largely upon the quality of the weak saliency model [41]. This motivates us to propose an integration mechanism (i.e. co-bootstrapping mechanism) which combines the strengths of various saliency methods to construct the weak saliency model.

Various methods have been developed for saliency estimation. These methods often have their highlights as well as weaknesses. A number of new methods have been proposed to deal with the fusion of saliency methods. Mai *et al.* [42] propose a saliency aggregation algorithm which combines saliency maps from various methods based on a data-driven approach. The contribution of each individual method is determined by a learned aggregation model. In [43], Le *et al.* discuss various aggregation methods. Self-adaptive weight is also exploited in the fusion of saliency methods. In [44], Cao *et al.* assign self-adaptively weights to each saliency map that participates in fusion process under the rank constraint. The aforementioned fusion methods mainly focus on obtaining the contribution (weight) of each individual method in fusion process. By contrast, we combine different saliency methods in a manner of bootstrap learning, which mines potential characteristics facilitating their complementary effects.

III. Bootstrap Saliency Model

Both weak and strong saliency models are exploited in the proposed bootstrap learning algorithm. Bootstrapping means that the learning process of the strong saliency model is bootstrapped with samples from the weak saliency model. In this paper, three weak saliency models are constructed based on original saliency model, co-map saliency model and co-sample saliency model respectively. The three weak models respectively correspond to *original bootstrapping*, *co-map bootstrapping* and *co-sample bootstrapping* processes. The original saliency model which based on image priors is the same as our previous work [41]. At first, we propose an original saliency map based on image priors. A graph cut method is used to smooth this coarse saliency map. More details will be introduced in Section III-B. The original bootstrapping results not only verify the effectiveness of the proposed framework but also show that the overall performance hinges on the quality of the weak saliency model. Therefore, we apply the co-bootstrapping mechanism to incorporate the proposed bootstrap learning algorithm with existing saliency methods, which explores complementary effects of various methods to construct the weak saliency model. Co-map saliency model and co-sample saliency model are explored in Section III-C1 and Section III-C2 respectively. The samples which are pertaining to the salient objects are considered as positive samples while those extracted from background are regarded as negative samples. Next, a strong classifier based on Multiple Kernel Boosting (MKB) [45] is learned to measure saliency where we

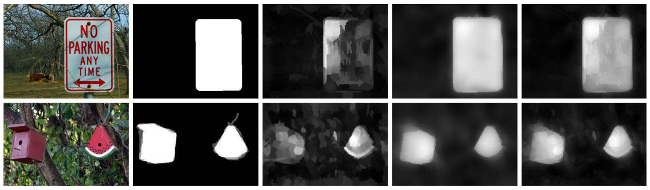


Fig. 1. Saliency maps generated by the proposed method. Brighter pixels indicate higher saliency values. Left to right: input, ground truth, weak saliency map, strong saliency map, and final saliency map.

use three feature descriptors in Section III-A and four kernels to exploit rich feature representations. Furthermore, multiscale superpixels are used in our work to detect salient objects of varying sizes. As the weak saliency model tends to detect fine details and the strong saliency model focuses on global shapes, we combine saliency maps from those two models to generate the final map. Figure 1 shows some saliency maps generated by the proposed method where brighter pixels indicate higher saliency values (the weak model is the original saliency model we proposed). The main steps of the proposed salient object detection algorithm has been shown in Figure 2.

A. Image Features

In vision tasks, superpixels have been used extensively as the basic units to capture the local structural information. Therefore, we compute a fixed number of superpixels from an input image using the Simple Linear Iterative Clustering (SLIC) method [46]. In this paper, three descriptors including the RGB, CIELab and Local Binary Pattern (LBP) features are used to describe each superpixel. For LBP feature, we consider it in a 3×3 neighborhood of each pixel. Next, each pixel is assigned to a value between 0 and 58 in the uniform pattern [47]. Then we construct an LBP histogram for each superpixel, i.e., a vector of 59 dimensions ($\{h_i\}, i = 1, 2, ...59$, where h_i is the value of the i-th bin in an LBP histogram).

B. Original Weak Saliency Model

In [48] and [49], the center-bias prior has been shown to be effective in salient object detection. Based on this assumption, we develop a method to construct a weak saliency model by exploiting the contrast between each region and the regions along the image border. However, existing contrast-based methods usually generate noisy results since low-level visual cues are limited. In this paper, we exploit the both center-bias and dark channel priors to better estimate saliency values.

The dark channel prior is proposed to remove the image haze [50]. The main observation is that, for regions that do not cover the sky (e.g., ground or buildings), there exist some pixels with low intensity values in one of the RGB color channels. From the above, the minimum pixel intensity in any such region is low. As shown in Figure 3, the dark channel of image patches is mainly generated by colored or dark objects and shadows, which usually appear in the salient regions. The sky region of an image usually belongs to the background, which is just consistent with the dark channel property for the sky region. Therefore, we exploit the dark channel property to estimate saliency of pixels. In addition, for situations where the input image has dark background or bright foreground, we use an adaptive weight computed based on the average value on the edge of dark channel map. If a patch centered at p has low intensity in a certain color channel, the patch likely belongs to salient regions, which means that p should be assigned a large saliency value. For a pixel p, the dark channel prior $S_d(p)$ is computed by

$$S_d(p) = 1 - \min_{q \in patch(p)} \left(\min_{ch \in \{r,g,b\}} \left(I^{ch}(q) \right) \right), \tag{1}$$

where $patch(p)$ is the 5×5 image patch centered at p and $I^{ch}(q)$ is the color value of pixel q on the corresponding color channel ch. Note that all the color values are normalized into $[0, 1]$. We achieve pixel-level accuracy instead of the patch-level counterpart in [50]. We also show the effect of dark channel prior quantitatively in Figure 9(b).

An input image is segmented into N superpixels, $\{c_i\}, i = 1, \ldots, N$. The regions along the image border are represented as $\{n_j\}, j = 1, \ldots, N_B$, where N_B is the number of regions along the image border. We compute the dark channel prior for each region c_i using $S_d(c_i) = \frac{1}{N_{c_i}} \sum_{p \in c_i} S_d(p)$, where N_{c_i} is the number of pixels within the region c_i. The coarse saliency value for the region c_i is constructed by

$$f_0(c_i) = g(c_i) \times S_d(c_i) \times \sum_{\kappa \in \{F_1, F_2, F_3\}} \left(\frac{1}{N_B} \sum_{j=1}^{N_B} d_\kappa(c_i, n_j) \right), \tag{2}$$

where $d_\kappa(c_i, n_j)$ is the Euclidean distance between region c_i and n_j in the feature space that κ represents, i.e., the RGB (F_1), CIELab (F_2) and LBP (F_3) texture features respectively. Note that all the distance values in each feature space are normalized into $[0, 1]$. In addition, $g(c_i)$ is computed based on the center prior using the normalized spatial distance between the center of the superpixel c_i and the image center [3]. Thus the saliency value of the region closer to the image center is assigned a higher weight. We generate a pixel-wise saliency map $\mathcal{M}_0$ using (2), where the saliency value of each superpixel is assigned to the contained pixels.

Most existing methods usually use Gaussian filtering to smooth saliency maps at the expense of accuracy. In this paper, we use a simple yet effective algorithm based on the Graph Cut method [51], [52], to determine the foreground and background regions in $\mathcal{M}_0$. In [53], Fine-grained (FG) and medium-grained (MG) segmentations are generated by the Graph Cut method to smooth the saliency map. It shows the effectiveness of Graph Cut method in smoothing. Our method does not need to generate segmentations first and the Graph Cut method is directly applied in the raw saliency map to generate a binary mask which is finally integrated with the raw saliency map.

Normally, there are two types of edges in the graph: *N-links* and *T-links*. N-links connect pairs of neighboring pixels. The weight of N-links corresponds to a penalty for discontinuity between the pixels. In contrast, T-links connect pixels with terminals. The weight of a T-link connecting a pixel and a terminal corresponds to a penalty for assigning the corresponding

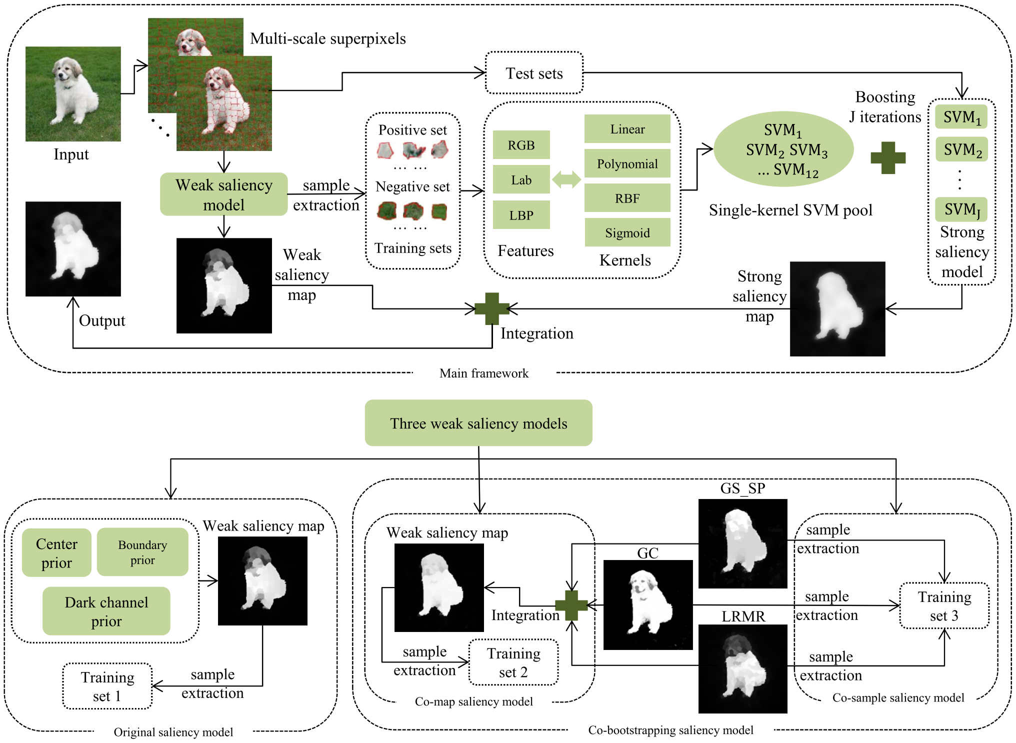


Fig. 2. Bootstrap learning for salient object detection. The weak saliency model is constructed to generate training samples for a strong model. A strong classifier based on multiple kernel boosting is learned to measure saliency where three feature descriptors are extracted and four kernels are used to exploit rich feature representations. The weak and strong saliency maps are weighted combined to generate the final saliency map. In this paper, three weak saliency models which respectively correspond to three bootstrapping processes are constructed.

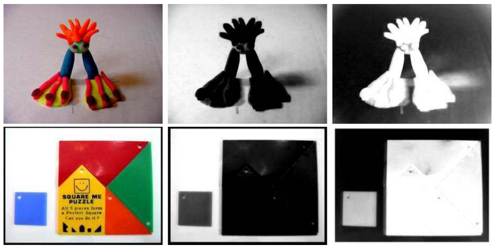


Fig. 3. Examples of dark channel prior. Left to right: input, dark channel map and dark channel prior (the opposite of dark channel map and brighter pixels indicate higher saliency values).

label to the pixel. In this paper, T-links are exploited to connect the pixel with foreground and background terminals. Given an input image, we construct an undirected graph $G = (V, E, T)$, where E is a set of undirected edges that connect the nodes V (pixels) while T is the set of the weights of nodes connected to the background and foreground terminals. The weight of each node (pixel) p connected to the foreground terminal is assigned with the saliency value in the pixel-wise map $\mathcal{M}_0$. Thus for each pixel p, the set T consists of two components, defined as $\{T^f(p)\}$ and $\{T^b(p)\}$, and is computed by

$$T^f(p) = \mathcal{M}_0(p), \quad T^b(p) = 1 - \mathcal{M}_0(p), \tag{3}$$

where $T^f(p)$ is the weight of pixel p connected to the foreground while $T^b(p)$ is the weight to the background. The minimum cost cut generates a foreground mask $\mathcal{M}_1$ using the Max-Flow [54] method to measure the probability of each pixel being foreground.

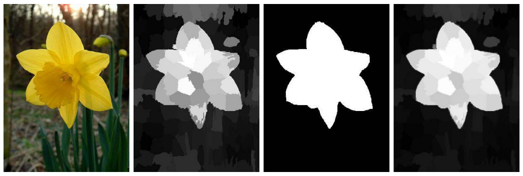


Fig. 4. Performance of Graph Cut. Left to right: input, saliency maps without Graph Cut, binary results using Graph Cut, saliency maps after summing up the previous two maps.

As shown in Figure 4, $\mathcal{M}_1$ is a binary map which may contain noise in both foreground and background. Thus we consider both the binary map $\mathcal{M}_1$ and the map $\mathcal{M}_0$ to construct the continuous and smoothed weak saliency map $\check{\mathcal{M}}_w$ by

$$\check{\mathcal{M}}_w = \frac{\mathcal{M}_0 + \mathcal{M}_1}{2}. \tag{4}$$

We show the performance of the Graph Cut method quantitatively in Figure 9(b).

C. Co-Bootstrapping Saliency Model

As mentioned in Section I, the overall performance of bootstrap learning algorithm depends largely upon the quality of the weak saliency model. If the weak saliency model fails to offer enough good training samples, the proposed algorithm is likely to fail on a specific image, as shown in Figure 5. To address the problem, the co-bootstrapping mechanism is proposed to combine the strengths of various existing saliency methods. Here we adopt co-map bootstrapping strategy to keep

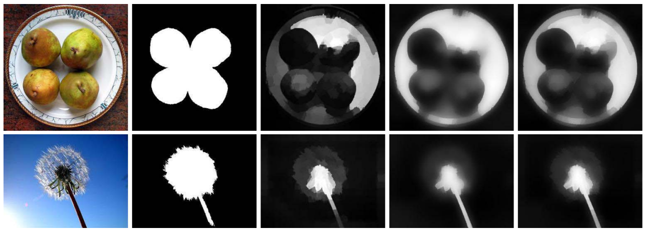


Fig. 5. Failure cases of the proposed algorithm as the weak saliency maps do not perform well. Left to right: input, ground truth, weak saliency map, strong saliency map and the bootstrap saliency map generated by the proposed algorithm.

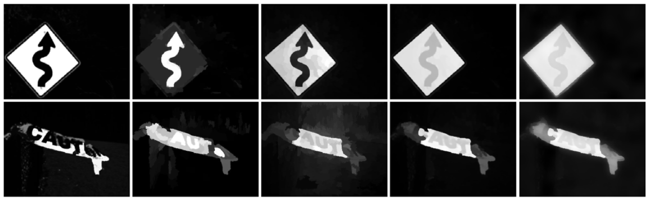


Fig. 6. Detection results of co-map saliency model. Left to right: GC saliency map, GS_SP saliency map, LRMR saliency map, combination of the previous three maps, our co-map bootstrapping results (all saliency maps are through graph cut process). There are superiorities as well as deficiencies of the three methods.

the integrity properties and co-sample bootstrapping strategy to mine the potential information of existing saliency methods.

1) Co-Map Saliency Model: In our work, we exploit saliency maps of three existing saliency detection approaches including GC [16], GS_SP [12], LRMR [14] to construct the weak saliency map, from which training samples will be extracted. In GC method [16], Cheng *et al.* utilize a soft abstraction method to remove unnecessary image details so that can effectively restrain the background. However, sometimes some important details of foreground are also removed in this method. Two kinds of background priors are applied in GS_SP method [16] to build a saliency model. It tends to find some accurate foreground pixels but not enough. In LRMR method [14], low-level features and high-level priors are combined together which leads to a uniform background. But we can find that the detected objects are not highlighted uniformly. Based on the above observations, we believe that any of the three methods have its superiorities as well as deficiencies. Therefore, it is reasonable to combine these methods together for better detection performance. To take full advantages of these methods, we linearly add saliency maps of them after Graph Cut process like in III-B. Thus a weak saliency map is computed by

$$\mathcal{M}_w = \frac{\mathcal{M}_{GC} + \mathcal{M}_{GS\_SP} + \mathcal{M}_{LRMR}}{3}. \tag{5}$$

where $\mathcal{M}_{GC}$, $\mathcal{M}_{GS\_SP}$, $\mathcal{M}_{LRMR}$ are graph cut saliency maps of GC, GS_SP, and LRMR respectively.

Some graph cut detection results of these three methods and our algorithm are shown in Figure 6. Just as we have mentioned before, our co-map bootstrapping method can combine the strengths of existing algorithms effectively.

2) Co-Sample Saliency Model: In Section III-C1, we utilize saliency maps of three methods by averaging them directly. It is able to keep the integrity property of each method while makes the results more dependent on the performance of every method. To reduce dependence, we propose a co-sample model which mines the intrinsic properties of three methods. It is reasonable to consider the property of each superpixel (sample) which may be salient in one method but non-salient in another one. We use samples which are directly extracted from graph cut saliency maps of three methods to bootstrap learning process. Compared with co-map model, the co-sample strategy can better explore the potential information of these methods. The number of extracted training samples are three times of the previous weak models. For final integration, we construct a weak saliency map $\mathcal{M}_w$ (not used for training samples generation) based on (5).

For the original saliency model and co-map saliency model, the training set for the strong classifier is selected from the weak saliency map. The training samples extraction mechanism (TSEM) for weak saliency map is as following: we compute the average saliency value for each superpixel and set two thresholds to generate the training set containing both positive and negative samples. The superpixels with saliency values larger than the high threshold are labeled as the positive samples with $+1$ while those with saliency values smaller than the low threshold as the negative samples labeled with -1. As for co-sample saliency model, saliency maps of three methods are all exploited for extraction of training samples. In other words, each saliency map of the three methods will be evaluated by TSME to generate training sets.

The three methods we exploit are all traditional saliency detection methods which mainly based on handcrafted features and proved to be complementary through our experiment. Recently, some new saliency methods (i.e. CNN-based approaches) are proposed and have favorable performances owing to massive training images. We also conduct our co-bootstrapping processes in CNN-based approaches. Saliency maps of MDF [35], MCDL [36] and DS [37] are exploited by our co-bootstrapping algorithms. It demonstrates that our co-bootstrapping methods can effectively improve not only traditional but also CNN-based methods. The results are displayed in Section IV.

D. Strong Saliency Model

One of the main difficulties using a Support Vector Machine (SVM) is to determine the appropriate kernel for the given dataset. This problem is more complicated when the dataset contains thousands of diverse images with different properties. While numerous saliency detection methods based on various features have been proposed, it is still not clear how these features can be well integrated. To cope with these problems, we present a method similar to the Multiple Kernel Boosting (MKB) [45] method to include multiple kernels of different features. We treat SVMs with different kernels as weak classifiers and then learn a strong classifier using the boosting method. Note that we restrict the learning process to each input image to avoid the heavy computational load of extracting features and learning kernels for a large amount of training data (as required in several discriminative methods [33] in the literature for saliency detection).

The MKB algorithm is a boosted Multiple Kernel Learning (MKL) method [55], which combines several SVMs of different kernels. For each image, we have the training samples $\{r_i, l_i\}_{i=1}^{H}$ from the weak saliency map $\check{\mathcal{M}}_w$ (See Section III-B) where r_i is the i-th sample, l_i represents the binary label of the sample and H indicates the number of the samples. The linear combination of kernels $\{k_m\}_{m=1}^{M}$ is defined by

$$k(r, r_i) = \sum_{m=1}^{M} \beta_m k_m(r, r_i), \quad \sum_{m=1}^{M} \beta_m = 1, \quad \beta_m \in \mathbb{R}_+, \quad (6)$$

where β_m is the kernel weight and M denotes the number of the weak classifiers, and $M = N_f \times N_k$. Here, N_f is the number of the features and N_k indicates the number of the kernels (e.g., $N_f = 3, N_k = 4$ in this work). For different feature sets, the decision function is defined as a convex combination,

$$Y(r) = \sum_{m=1}^{M} \beta_m \sum_{i=1}^{H} \alpha_i l_i k_m(r, r_i) + \bar{b}, \quad (7)$$

where α_i is the Lagrange multiplier while $\bar{b}$ is the bias in the standard SVM algorithm. The parameters $\{\alpha_i\}$, $\{\beta_m\}$ and $\bar{b}$ can be learned from a joint optimization process.

We note that (7) is a conventional function for the MKL method. In this paper we use the boosting algorithm instead of the simple combination of single-kernel SVMs in the MKL method. We rewrite (7) as

$$Y(r) = \sum_{m=1}^{M} \beta_m (\boldsymbol{\alpha}^\top \mathbf{k}_m(r) + \bar{b}_m), \quad (8)$$

where $\boldsymbol{\alpha} = [\alpha_1 l_1, \alpha_2 l_2, \ldots, \alpha_H l_H]^\top$, $\mathbf{k}_m(r) = [k_m(r, r_1), k_m(r, r_2), \ldots, k_m(r, r_H)]^\top$ and $\bar{b} = \sum_{m=1}^{M} \bar{b}_m$. By setting the decision function of a single-kernel SVM as $z_m(r) = \boldsymbol{\alpha}^\top \mathbf{k}_m(r) + \bar{b}_m$, the parameters can be learned straightforwardly. Thus, (8) can be rewritten as

$$Y(r) = \sum_{j=1}^{J} \beta_j z_j(r). \quad (9)$$

In order to compute the parameters β_j, we use the Adaboost method and the parameter J in (9) denotes the number of iterations of the boosting process. We consider each SVM as a weak classifier and the final strong classifier $Y(r)$ is the weighted combination of all the weak classifiers. Starting with uniform weights, $\omega_1(i) = 1/H, i = 1, 2, \ldots, H$, for the SVM classifiers, we obtain a set of decision functions $\{z_m(r)\}, m = 1, 2, \ldots, M$. At the j-th iteration, we compute the classification error for each of the weak classifiers,

$$\epsilon_m = \frac{\sum_{i=1}^{H} \omega(i)|z_m(r_i)|(\mathrm{sgn}(-l_i z_m(r_i)) + 1)/2}{\sum_{i=1}^{H} \omega(i)|z_m(r_i)|}, \quad (10)$$

where sgn(x) is the sign function, which equals to 1 when $x > 0$ and -1 otherwise. We locate the decision function $z_j(r)$ with the minimum error ϵ_j, i.e., $\epsilon_j = \min_{1 \le m \le M} \epsilon_m$. Then the combination coefficient β_j is computed by $\beta_j = \frac{1}{2} \log \frac{1-\epsilon_j}{\epsilon_j} \cdot \frac{1}{2}(\mathrm{sgn}(\log \frac{1-\epsilon_j}{\epsilon_j}) + 1)$. Note that β_j must be larger than 0, indicating $\epsilon_j < 0.5$, which accords with the basic hypothesis that the boosting method could make the weak classifiers into a strong one. In addition, we update the weight using the following equation,

$$\omega_{j+1}(i) = \frac{\omega_j(i) e^{-\beta_j l_i z_j(r_i)}}{2\sqrt{\epsilon_j(\epsilon_j - 1)}}. \quad (11)$$

After J iterations, all the β_j and $z_j(r)$ are computed and we have a boosted classifier (9) as the saliency model learned directly from an input image. We apply this strong saliency model to the test samples (based on all the superpixels of an input image), and a pixel-wise saliency map is thus generated.

To improve the accuracy of the map, we first use the Graph Cut method to smooth the saliency detection results. Next, we obtain the strong saliency map $\check{\mathcal{M}}_s$ by further enhancing the saliency map with the guided filter [56] as it has been shown to perform well as an edge-preserving smoothing operator.

E. Multiscale Saliency Maps

The accuracy of the saliency map is sensitive to the number of superpixels as salient objects are likely to appear at different scales. To deal with the scale problem, we generate four layers of superpixels with different granularities, where $N = 100, 150, 200, 250$ respectively. For Section III-B, we represent the weak saliency map at each scale as $\{\check{\mathcal{M}}_{w_i}\}$ and the multiscale weak saliency map is computed by $\mathcal{M}_w = \frac{1}{4} \sum_{i=1}^{4} \check{\mathcal{M}}_{w_i}$. Next, the training sets from the multi-scales are used to train one strong saliency model and the test sets (based on all the superpixels from multi-scales) are tested by the learned model simultaneously. Four strong saliency maps from four scales are constructed (See Section III-D), denoted as $\{\check{\mathcal{M}}_{s_i}\}, i = 1, 2, 3, 4$. Finally, we obtain the final strong saliency map as $\mathcal{M}_s = \frac{1}{4} \sum_{i=1}^{4} \check{\mathcal{M}}_{s_i}$. As such, the proposed method is robust to scale variation. For co-bootstrapping models (See Section III-C1 and Section III-C2), we construct the weak saliency map $\mathcal{M}_w$ based on (5) and only two scales are introduced into our experiment (where $N = 100, 150$) for efficiency of our algorithm. Then the final strong map is obtained by $\mathcal{M}_s = \frac{1}{2} \sum_{i=1}^{2} \check{\mathcal{M}}_{s_i}$.

F. Integration

The proposed weak and strong saliency maps have complementary properties. The weak map is likely to detect fine details and to capture local structural information due to the contrast-based measure. In contrast, the strong map works well by focusing on global shapes for most images except the case when the test background samples have similarity with the positive training set or large differences compared to the negative training set, or vice versa for the test foreground sample. In this case, the strong map may mis-classify the test regions as shown in the bottom row of Figure 1. Thus we integrate these two maps by a weighted combination,

$$\mathcal{M} = \sigma \mathcal{M}_s + (1 - \sigma) \mathcal{M}_w, \quad (12)$$

where σ is a balance factor for the combination, and $\sigma = 0.7$ to weigh the strong map more than the weak map, and $\mathcal{M}$ is the final saliency map via bootstrap learning. To better show the performance of our co-bootstrapping models, the values of σ will be modified to 0.5 in co-map and co-sample parts.

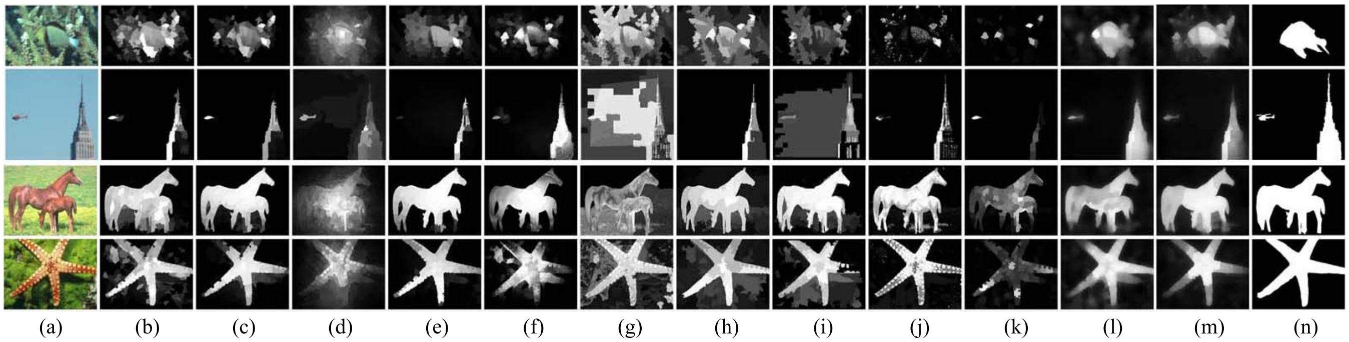


Fig. 7. Comparison of our saliency maps with ten state-of-the-art methods. Left to right: (a) input (b) GS_SP [12] (c) wCO [20] (d) LRMR [14] (e) GMR [18] (f) DSR [19] (g) XL13 [13] (h) HS [17] (i) RC-J [15] (j) GC [16] (k) SF [11] (l) Original-bootstrapped (m) Co-bootstrapped (n) ground truth. Our model is able to detect both the foreground and background uniformly.

TABLE I

AUC (Area Under ROC Curve) on the ASD, SED2, SOD, THUS, Pascal-S and DUT-OMRON Data Sets. The Best Two Results Are Shown in Red and Blue Fonts Respectively. The Colomn Named "Ori-b" Denotes the Original Model After Bootstrapping Using the Proposed Approach. The Proposed Co-Bootstrapping Methods Named "Co-Sample" and "Co-Map" Rank First and Second on the Six Data Sets. The Two Rows Named "*ASD (b)*" Show the AUC of the Saliency Results by Taking Other State-of-the-Art Saliency Maps as the Weak Saliency Maps in the Proposed Approach on the ASD Dataset. All the Evaluation Results of the State-of-the-Art Methods Are Largely Improved Over the Original Results as Shown in the Two Rows Named "*ASD*"

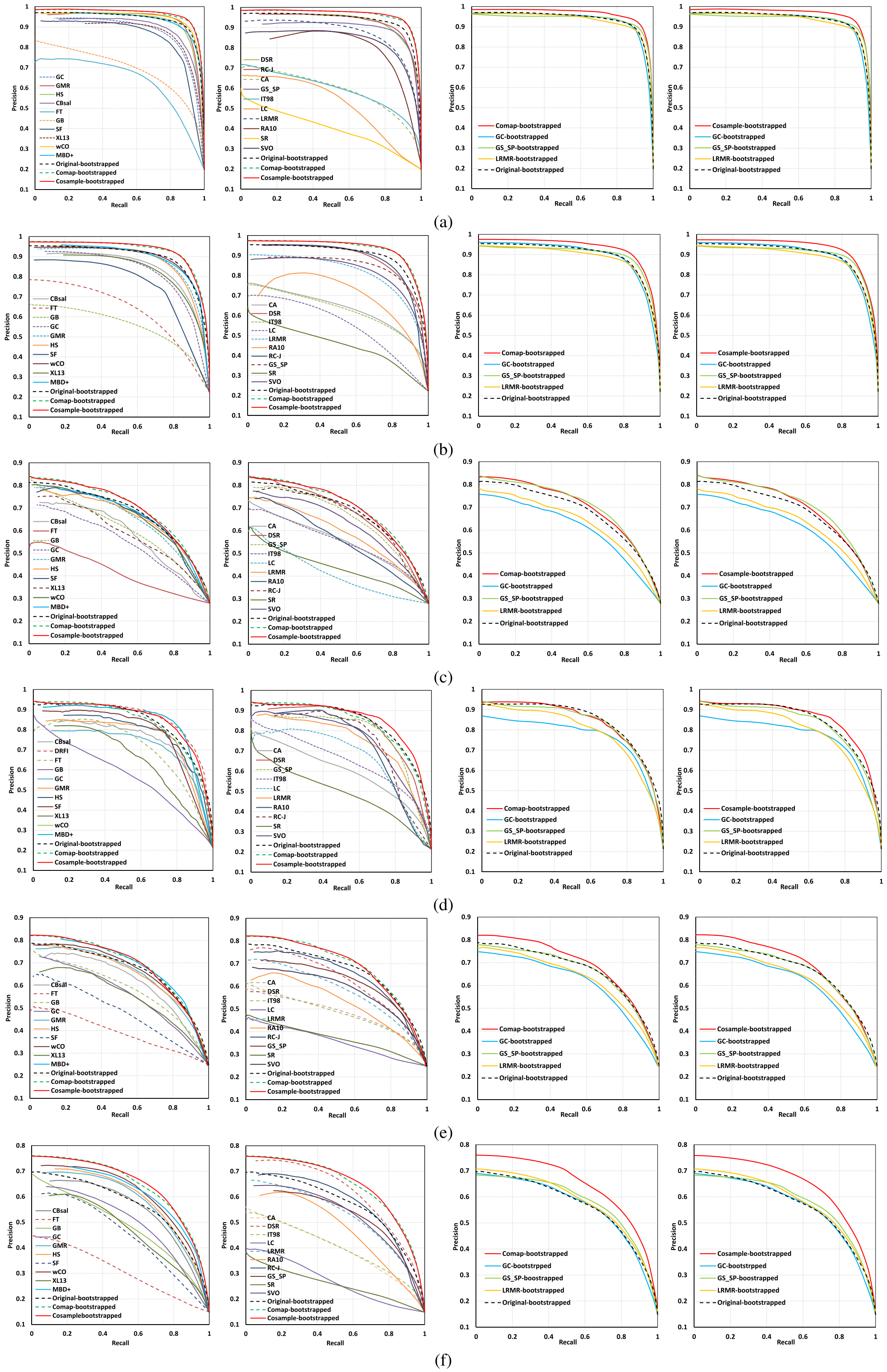
	Co-sample	C-omap	Ori-b	HS	RC-J	GC	DSR	GS_SP	GMR	SF	XL13	CBsal
ASD	**.9898**	.9890	.9828	.9683	.9735	.9456	.9774	.9754	.9700	.9233	.9609	.9628
SED2	**.9413**	.9226	**.9363**	.8387	.8606	.8618	.9136	.8999	.8620	.8501	.8470	.8728
SOD	**.8498**	**.8490**	.8477	.8169	.8238	.7181	.8380	.7982	.7982	.8238	.7868	.7409
THUS	**.9730**	**.9718**	.9635	.9322	.9364	.9032	.9504	.9462	.9390	.8510	.9353	.9270
Pascal-S	**.8696**	**.8686**	.8682	.8368	.8379	.7479	.8299	.8553	.8315	.6830	.7983	.8087
DUT-OMRON	**.9114**	**.9051**	.8794	.8604	.8592	.7931	.8922	.8786	.8500	.7628	.8160	.8419
ASD (b)	-	-	-	.9876	.9869	.9773	.9872	.9888	.9844	.9723	.9791	.9811
	DRFI	wCO	MBD+	LRMR	RA10	SVO	GB	FT	CA	SR	LC	IT98
ASD	-	.9805	.9788	.9593	.9326	.9530	.9146	.8375	.8736	.6973	.7772	.8738
SED2	.9349	.9062	.9028	.8886	.8500	.8773	.8448	.8185	.8585	.7593	.8366	.8904
SOD	-	.8217	.8358	.7810	.7710	.8043	.8191	.6006	.7868	.6695	.6168	.7862
THUS	-	.9525	.9499	.9199	.8810	.9280	.8132	.7890	.8712	.7149	.7673	.8655
Pascal-S	-	.8597	.8598	.8121	.7836	.8226	.8380	.6220	.7829	.6585	.6191	.7797
DUT-OMRON	-	.8927	.8930	.8566	.8264	.8662	.8565	.6758	.8137	.6799	.6549	.8218
ASD (b)	-	**.9904**	.9861	.9825	.9817	.9722	.9619	.9506	.9531	.8530	.8988	.9479

IV. Experimental Results

For traditional methods, we present experimental results of 23 saliency detection methods including the proposed algorithms on six benchmark data sets (ASD, THUS, SOD, SED2, Pascal-S, DUT-OMRON). While for CNN-based methods, the experiments results on four benchmark data sets (SOD, Pascal-S, DUT-OMRON, ECSSD) are displayed for the reason that CNN-based approaches all achieve excellent performance on simple data sets like ASD. The SOD dataset [57] is composed of 300 images from the Berkeley segmentation dataset. Some of the images in the SOD dataset include more than one salient object. The SED2 dataset [58] contains 100 images. It is challenging due to the fact that every image has two salient objects. The Pascal-S dataset [59] contains 850 images. The DUT-OMRON dataset [18] contains 5168 challenging images. The ECSSD dataset [17] is composed of 1, 000 structurally complex images acquired from the Internet. All images in these data sets correspond to manually labeled ground truth. The experiments are carried out using MATLAB on a desktop computer with an Intel i7−3770 CPU (3.4 GHz) and 32GB RAM. For fair comparison, we use the original source code or the provided saliency detection results in the literature.

For traditional methods, we evaluate the proposed algorithms and other 20 state-of-the-art methods including the IT98 [1], SF [11], LRMR [14], wCO [20], GS_SP [12], XL13 [13], RA10 [2], GB [9], LC [10], SR [6], FT [5], CA [8], SVO [7], CBsal [3], GMR [18], GC [16], HS [17], RC-J [15], DSR [19], and MBD+ [60]methods on the ASD, THUS, SOD, SED2, Pascal-S and DUT-OMRON data sets. In addition, the DRFI [33] method uses images and ground truth for training, which contains part of the ASD, THUS and SOD data sets, and the results on the Pascal-S dataset are not provided. Accordingly, we only compare our method with the DRFI model on the SED2 dataset. Therefore, our methods are evaluated with 21 methods on the SED2 data sets. The MSRA [48] dataset consists of 5,000 images. Since more than 3,700 images in the MSRA dataset are included in the THUS dataset, we do not present the evaluation results on this dataset due to space limitations.

For CNN-based methods, we evaluate the proposed co-bootstrapping algorithms and other 4 state-of-the-art methods including the LEGS [34], MDF [35], MCDL [36] and DS [37] methods on the the SOD, Pascal-S, DUT-OMRON and ECSSD data sets. In LEGS [34], 340 images from the



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 8. P-R curve results on six data sets. (a) ASD dataset. (b) THUS dataset. (c) SOD dataset. (d) SED2 dataset. (e) Pascal-S dataset. (f) DUT-OMRON dataset.

Pascal-S dataset are used to train the networks while 100 images from the SOD dataset are exploited for training process in MDF [35]. Therefore, we only test the remaining images of these two data sets in our experiment for fair comparison.

A. Qualitative Results

We present some results of saliency maps generated by twelve methods for qualitative comparison in Figure 7, where “Original-bootstrapped” and “Co-bootstrapped” mean the orig-

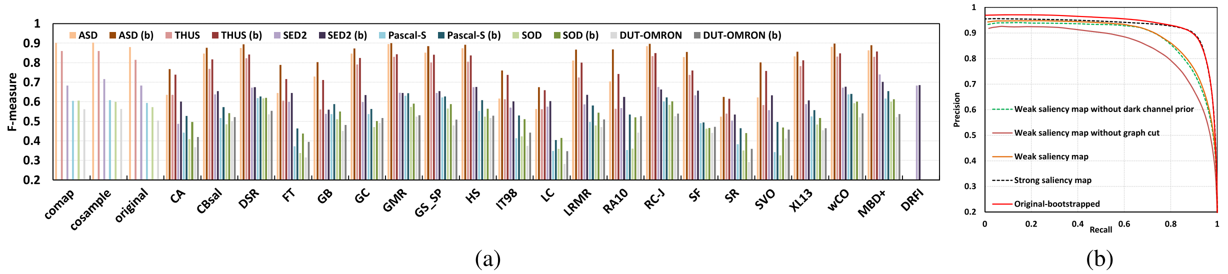


Fig. 9. (a) is the F-measure values of 21 methods on six data sets. Note that " * (b)" shows improvement of state-of-the-art methods by the bootstrap learning approach on the corresponding dataset as stated in Section IV-D1. (b) shows performance of each component in the proposed method on the ASD dataset.

TABLE II

AVERAGE F-MEASURE AND AUC (AREA UNDER ROC CURVE) ON THE SOD, PASCAL-S, DUT-OMRON AND ECSSD DATA SETS. THE BEST TWO RESULTS ARE SHOWN IN **red** AND **blue** FONTS RESPECTIVELY. NOTE THAT " * *-b*" SHOWS THE BOOTSTRAP LEARNING RESULTS USING MAPS OF " * " AS WEAK SALIENCY MAPS.THE PROPOSED CO-BOOTSTRAPPING METHODS NAMED "CO-SAMPLE" AND "CO-MAP" RANK FIRST AND SECOND ON THE FOUR DATA SETS

Data sets		Co-sample	Co-map	LEGS	LEGS-*b*	MDF	MDF-*b*	MCDL	MCDL-*b*	DS	DS-*b*
SOD	F-Measure	**.6895**	**.6854**	.6517	.6485	.6648	.6616	.6459	.6614	.6749	.6779
	AUC	**.9221**	**.9162**	.8118	.8431	.8553	.8930	.7984	.8640	.9071	.9101
Pascal-S	F-Measure	**.7173**	**.7089**	.6973	.6915	.7071	.6910	.6893	.6976	.6574	.6750
	AUC	**.9460**	**.9407**	.8806	.9008	.8321	.8936	.8601	.9083	.9312	.9353
DUT-OMRON	F-Measure	**.6492**	**.6428**	.5915	.5934	.5972	.6173	.6238	.6335	.6028	.6212
	AUC	**.9616**	**.9590**	.8839	.9033	.8963	.9274	.9007	.9373	.9453	.9494
ECSSD	F-Measure	.8313	**.8320**	.7886	.7928	.8049	.7932	.7953	.8098	.8261	**.8350**
	AUC	**.9752**	**.9740**	.9226	.9416	.8970	.9420	.9157	.9561	.9654	.9707

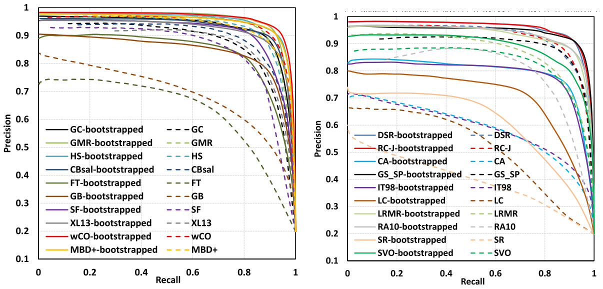
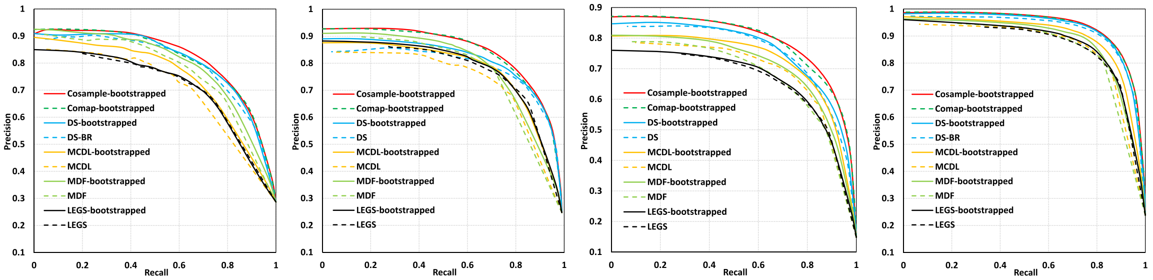


Fig. 10. P-R curve results show improvement of state-of-the-art methods by the bootstrap learning approach on the ASD dataset.

inal weak saliency model and co-map model bootstrapped by the proposed learning approach respectively (Results of co-map are similiar to co-sample thus not displayed). The saliency maps generated by the proposed algorithms highlight the salient objects well with fewer noisy results. We note that these salient objects appear at different image locations although the center-bias is used in the proposed algorithm. The detected foreground and background in our maps are smooth due to the using of the Graph Cut and guided filtering methods. As a result of using both weak and strong saliency maps, the proposed bootstrap learning algorithm performs well for images containing multiple objects as shown in the second and third rows of Figure 7. Furthermore, due to the contribution of the LBP features (effective for texture classification), the proposed method is able to detect salient objects accurately despite similar appearance to the background regions as shown in the first row of Figure 7.

B. Quantitative Results

We use the Precision and Recall (P-R) curve to evaluate all the methods. As mentioned in Section III-C, massive training images are involved in CNN-based approaches while traditional methods usually just based on handcrafted features. Therefore, the experimental comparisons are divided into traditional methods and CNN-based methods. For traditional methods, Figure 8 shows the P-R curves where several state-of-the-art methods and the proposed algorithms perform well. Saliency detection results of co-bootstrapping models of traditional methods on six data sets are also evaluated by P-R curve. For CNN-based methods, the P-R results of several state-of-the-art methods and the proposed co-bootstrapping algorithms on four data sets are displayed in Figure 11. To better assess these methods, we compute the Area Under ROC Curve (AUC) for the best performing methods. Table I shows that the proposed algorithms perform favorably against other state-of-the-art methods in terms of AUC on all the six data sets that contain both single and multiple salient objects. In Table II, we show the AUC reults of CNN-based methods and our co-bootstrapping algorithms on four data sets. It demonstrates that the CNN-based methods can be improved by our algorithm individually and the co-bootstrapping algorithms have the best results. In addition, we measure the quality of the saliency maps using the F-Measure by adaptively setting a segmentation threshold for binary segmentation [5]. Figure 9(a) shows the F-Measure values of the evaluated traditional methods on the six data sets. We also show the F-Measure values of CNN-based methods and our co-bootstrapping algorithms



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

Fig. 11. P-R curve results on four data sets. (a) SOD. (b) Pascal-S. (c) DUT-OMRON. (d) ECSSD.

in Table II. Overall, the proposed algorithms perform well against the state-of-the-art methods.

C. Analysis of the Bootstrap Saliency Model

Every component in the proposed algorithm contributes to the final saliency map. Figure 9(b) shows the performance of each step in the proposed method, i.e., the dark channel prior, graph cut, weak saliency map, and strong saliency map, among which the dark channel prior appears to contribute least but is still indispensable for the overall performance. The proposed original weak saliency model may generate less accurate results than several state-of-the-art methods, but it is efficient with less computational complexity.

D. Bootstrapping Algorithm Measurement

As mentioned in section I, our bootstrap learning algorithm can be easily used by existing bottom-up methods for performance improvement. In addition, the proposed method provides us with an effective way to combine the advantages of different saliency detection approaches which complement each other for better performance. We propose two different kinds of bootstrapping strategies to combine the strengths of various methods: *co-map bootstrapping* and *co-sample bootstrapping*. In co-map strategy, we exploit saliency maps of three existing saliency detection approaches by adding them together to keep the integrity property of each method. While in co-sample one, salient and non-salient samples are extracted directly from saliency maps of three methods which can better explore the potential information of these methods. In IV-D1, we show the performance of bootstrapping methods to improve the existing saliency methods. The performance of co-bootstrapping algorithms are shown in IV-D2.

1) Bootstrapping State-of-the-Art Methods: The proposed bootstrap learning algorithm can be easily applied to existing saliency methods to improve their performance. We generate different weak saliency maps by applying the graph cut method on the results generated by the state-of-the-art methods. Note that we only use two scales instead of four scales for efficiency and use equal weights in (12) (to better use these “weak” saliency maps) in the experiments. Figure 10 shows the P-R curves on the ASD dataset and Figure 9(a) shows the F-measure on six tested data sets. In addition, the AUC measures are shown on the two rows named *“ASD (b)”* of Table I. These results show that the performance of all state-of-the-art methods can be significantly improved by the proposed bootstrap learning algorithm.

2) Co-Bootstrapping Measurement: Like IV-D1, we use two scales instead of four scales for efficiency and use equal weights in (12) in the experiments. In Figure 8, we show our co-bootstrapping P-R results of three traditional saliency methods. The co-map bootstrapped results perform favorably against the state-of-the-art methods although detection results generated by each of the three methods are not so excellent. We also compare one-map bootstrapping (i.e.GC bootstrapped, GS_SP bootstrapped and LRMR bootstrapped) and co-map bootstrapping results to evaluate the effectiveness of of co-map model. It demonstrates that our co-map model can effectively integrate superiorities of three methods together to get better performance. Different from co-map model, we directly extract training samples from saliency maps of three methods to explore the potential information of these methods. The co-sample bootstrapped results also perform well against the state-of-the-art methods. Just as in IV-D2, we also compare co-sample results with one-map results for performance evaluation.

Furthermore, we also conduct our co-bootstrapping processes in CNN-based methods. Three CNN-based methods (MDF [35], MCDL [36] and DS [37]) are exploited to get the co-bootstrapping results. The P-R results are displayed in Figure 11 and the F-Measure and AUC results are listed in Table II. It shows that our co-bootstrapping models can effectively improve both traditional and CNN-based methods. The detection performances of co-map and co-sample model are similar to each other, which demonstrates that both of the two bootstrapping models are reasonable and we can consider integration of different methods from integrity perspective and potential information perspective.

V. CONCLUSION

In this paper, we propose a bootstrap learning model for salient object detection in which both weak and strong saliency models are constructed and integrated. Our learning process is restricted within multiple scales of the input image and is unsupervised since the training examples for the strong model are determined by a weak saliency map based on contrast and image priors. The strong saliency model is constructed based on the MKB algorithm which combines all the weak classifiers into a strong one using the Adaboost algorithm. Extensive experimental results demonstrate that the proposed approach performs favorably against the state-of-the-art methods. In addition, the proposed bootstrap learning algorithm can be applied to other saliency models for significant improvement. We also prove that saliency maps of different methods have complementary effects and the co-bootstrapping methods effectively combine the advantages of all the methods.

References

[1] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[2] E. Rahtu, J. Kannala, M. Salo, and J. Heikkilä, "Segmenting salient objects from images and videos," in *Proc. ECCV*, 2010, pp. 366–379.

[3] H. Jiang, J. Wang, Z. Yuan, T. Liu, N. Zheng, and S. Li, "Automatic salient object segmentation based on context and shape prior," in *Proc. BMVC*, 2011, pp. 110.1–110.12.

[4] D. A. Klein and S. Frintrop, "Center-surround divergence of feature statistics for salient object detection," in *Proc. ICCV*, Nov. 2011, pp. 2214–2219.

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

[6] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *Proc. CVPR*, Jun. 2007, pp. 1–8.

[7] K.-Y. Chang, T.-L. Liu, H.-T. Chen, and S.-H. Lai, "Fusing generic objectness and visual saliency for salient object detection," in *Proc. ICCV*, Nov. 2011, pp. 914–921.

[8] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," in *Proc. CVPR*, Jun. 2010, pp. 2376–2383.

[9] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Proc. NIPS*, 2006, pp. 545–552.

[10] Y. Zhai and M. Shah, "Visual attention detection in video sequences using spatiotemporal cues," in *Proc. ACM MM*, 2006, pp. 815–824.

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

[12] Y. Wei, F. Wen, W. Zhu, and J. Sun, "Geodesic saliency using background priors," in *Proc. ECCV*, 2012, pp. 29–42.

[13] Y. Xie, H. Lu, and M.-H. Yang, "Bayesian saliency via low and mid level cues," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1689–1698, May 2013.

[14] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," in *Proc. CVPR*, Jun. 2012, pp. 853–860.

[15] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, Mar. 2015.

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

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

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

[19] X. Li, H. Lu, L. Zhang, X. Ruan, and M.-H. Yang, "Saliency detection via dense and sparse reconstruction," in *Proc. ICCV*, Dec. 2013, pp. 2976–2983.

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

[21] N. Tong, H. Lu, Y. Zhang, and X. Ruan, "Salient object detection via global and local cues," *Pattern Recognit.*, vol. 48, no. 10, pp. 3258–3267, Oct. 2015, doi: 10.1016/j.patcog.2014.12.005.

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

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

[24] J. Sun, H. Lu, and S. Li, "Saliency detection based on integration of boundary and soft-segmentation," in *Proc. ICIP*, Sep./Oct. 2012, pp. 1085–1088.

[25] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang, "Saliency detection via absorbing Markov chain," in *Proc. ICCV*, Dec. 2013, pp. 1665–1672.

[26] W. Zou, K. Kpalma, Z. Liu, and J. Ronsin, "Segmentation driven low-rank matrix recovery for saliency detection," in *Proc. 24th Brit. Mach. Vis. Conf. (BMVC)*, Sep. 2013, pp. 1–14,

[27] Z. Liu, W. Zou, and O. Le Meur, "Saliency tree: A novel saliency detection framework," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 1937–1952, May 2013.

[28] W. Zou and N. Komodakis, "HARF: Hierarchy-associated rich features for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 406–414.

[29] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 110–119.

[30] Y. Kong, L. Wang, X. Liu, H. Lu, and X. Ruan, "Pattern mining saliency," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 583–598.

[31] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell, "SUN: A Bayesian framework for saliency using natural statistics," *J. Vis.*, vol. 8, no. 7, p. 32, 2008.

[32] J. Yang and M.-H. Yang, "Top-down visual saliency via joint CRF and dictionary learning," in *Proc. CVPR*, Jun. 2012, pp. 2296–2303.

[33] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *Proc. CVPR*, 2013, pp. 2083–2090.

[34] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3183–3192.

[35] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5455–5463.

[36] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1265–1274.

[37] X. Li *et al.*, "DeepSaliency: Multi-task deep neural network model for salient object detection," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3919–3930, Aug. 2016.

[38] T. Wang, L. Zhang, H. Lu, C. Sun, and J. Qi, "Kernelized subspace ranking for saliency detection," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 450–466.

[39] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Saliency detection with recurrent fully convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 825–841.

[40] B. Kuipers and P. Beeson, "Bootstrap learning for place recognition," in *Proc. AAAI*, 2002, pp. 1–7.

[41] N. Tong, H. Lu, X. Ruan, and M.-H. Yang, "Salient object detection via bootstrap learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1884–1892.

[42] L. Mai, Y. Niu, and F. Liu, "Saliency aggregation: A data-driven approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 1131–1138.

[43] O. Le Meur and Z. Liu, "Saliency aggregation: Does unity make strength?" in *Proc. Asian Conf. Comput. Vis.*, 2014, pp. 18–32.

[44] X. Cao, Z. Tao, B. Zhang, H. Fu, and W. Feng, "Self-adaptively weighted co-saliency detection via rank constraint," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 4175–4186, Sep. 2014.

[45] F. Yang, H. Lu, and Y.-W. Chen, "Human tracking by multiple kernel boosting with locality affinity constraints," in *Proc. ACCV*, 2010, pp. 39–50.

[46] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels," École Polytechnique Fédérale Lausanne, Lausanne, Switzerland, Tech. Rep. 149300, 2010.

[47] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[48] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," in *Proc. CVPR*, Jun. 2007, pp. 1–8.

[49] A. Borji, D. N. Sihite, and L. Itti, "Salient object detection: A benchmark," in *Proc. ECCV*, 2012, pp. 414–429.

[50] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2341–2353, Dec. 2011.

[51] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.

[52] V. Kolmogorov and R. Zabin, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, Feb. 2004.

[53] W. Zou, Z. Liu, K. Kpalma, J. Ronsin, Y. Zhao, and N. Komodakis, "Unsupervised joint salient region detection and object segmentation," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3858–3873, Nov. 2015.

[54] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.

[55] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan, “Multiple kernel learning, conic duality, and the SMO algorithm,” in *Proc. ICML*, 2004, pp. 1–6.

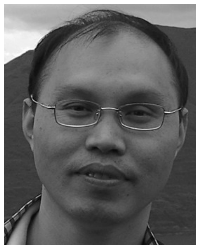
[56] K. He, J. Sun, and X. Tang, “Guided image filtering,” in *Proc. ECCV*, 2010, pp. 1–14.

[57] V. Movahedi and J. H. Elder, “Design and perceptual validation of performance measures for salient object segmentation,” in *Proc. CVPRW*, Jun. 2010, pp. 49–56.

[58] S. Alpert, M. Galun, R. Basri, and A. Brandt, “Image segmentation by probabilistic bottom-up aggregation and cue integration,” in *Proc. CVPR*, Jun. 2007, pp. 1–8.

[59] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. Yuille, “The secrets of salient object segmentation,” in *Proc. CVPR*, 2014, pp. 280–287.

[60] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, and R. Mech, “Minimum barrier salient object detection at 80 FPS,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1404–1412.

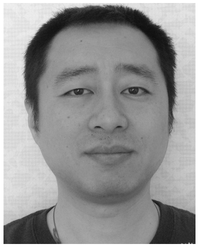


Huchuan Lu (SM’12) received the M.Sc. degree in signal and information processing and the Ph.D. degree in system engineering from the Dalian University of Technology (DUT), Dalian, China, in 1998 and 2008, respectively. He joined DUT in 1998 as a Faculty Member, where he is currently a Full Professor with the School of Information and Communication Engineering. His current research interests include computer vision and pattern recognition with a focus on visual tracking, saliency detection, and segmentation. He is a member of the Association for Computing Machinery and an Associate Editor of the IEEE TRANSACTIONS ON CYBERNETICS.

Xiaoning Zhang received the B.E. degree in electronic information engineering from the Dalian University of Technology (DUT), Dalian, China, in 2015. She is currently pursuing the master’s degree with the School of Information and Communication Engineering, DUT. Her research interest is in saliency detection.

Jinqing Qi (M’14) received the Ph.D. degree in communication and integrated system from the Tokyo Institute of Technology, Tokyo, Japan, in 2004. He is currently an Associate Professor of Information and Communication Engineering with the Dalian University of Technology, Dalian, China. His recent research interests focus on computer vision, pattern recognition, and machine learning.

Na Tong received the B.E. degree in electronic information engineering and the M.S. degree in signal and information processing from the Dalian University of Technology (DUT), Dalian, China, in 2012 and 2015, respectively. Her research interest is in saliency detection.



Xiang Ruan received the B.E. degree from Shanghai Jiao Tong University, Shanghai, China, in 1997, and the M.E and Ph.D. degrees from Osaka City University, Osaka, Japan, in 2001 and 2004, respectively. He is currently the CEO and a Co-founder of Tiwaki Company, Japan. His current research interests include computer vision, machine learning, and image processing.

Ming-Hsuan Yang (SM’06) received the Ph.D. degree in computer science from the University of Illinois at Urbana–Champaign, Urbana, in 2000. He was a Senior Research Scientist with the Honda Research Institute, where he was involved in vision problems related to humanoid robots. He is currently an Assistant Professor with the Department of Electrical Engineering and Computer Science, University of California at Merced, Merced. He has co-authored the book *Face Detection and Gesture Recognition for Human-Computer Interaction* (Kluwer, 2001). He served as an Editor of the Special Issue on face recognition for *Computer Vision and Image Understanding* in 2003.

Dr. Yang is a Senior Member of the ACM. He was a recipient of the Ray Ozzie fellowship for his research work in 1999. He received the Natural Science Foundation CAREER Award in 2012, the Campus Wide Senate Award for Distinguished Early Career Research at UC in 2011, and the Google Faculty Award in 2009. He serves as an Area Chair of the IEEE International Conference on Computer Vision in 2011, the IEEE Conference on Computer Vision and Pattern Recognition in 2008 and 2009, the Asian Conference on Computer in 2009, 2010, and 2012. He served as an Associate Editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE from 2007 to 2011, and the *Image and Vision Computing*. He served as an Editor of the Special Issue on real world face recognition of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.