Learning Discriminative Subspaces on Random Contrasts for Image Saliency Analysis

Shu Fang, Jia Li, *Senior Member, IEEE*, Yonghong Tian, *Senior Member, IEEE*, Tiejun Huang, *Senior Member, IEEE*, and Xiaowu Chen, *Senior Member, IEEE*

***Abstract*— In visual saliency estimation, one of the most challenging tasks is to distinguish targets and distractors that share certain visual attributes. With the observation that such targets and distractors can sometimes be easily separated when projected to specific subspaces, we propose to estimate image saliency by learning a set of discriminative subspaces that perform the best in popping out targets and suppressing distractors. Toward this end, we first conduct principal component analysis on massive randomly selected image patches. The principal components, which correspond to the largest eigenvalues, are selected to construct candidate subspaces since they often demonstrate impressive abilities to separate targets and distractors. By projecting images onto various subspaces, we further characterize each image patch by its contrasts against randomly selected neighboring and peripheral regions. In this manner, the probable targets often have the highest responses, while the responses at background regions become very low. Based on such random contrasts, an optimization framework with pairwise binary terms is adopted to learn the saliency model that best separates salient targets and distractors by optimally integrating the cues from various subspaces. Experimental results on two public benchmarks show that the proposed approach outperforms 16 state-of-the-art methods in human fixation prediction.**

***Index Terms*— Fixation prediction, learning-based model, random contrast, subspace analysis, visual saliency.**

I. Introduction

VISUAL attention, which is one of the most important mechanisms in the human vision system, works like a filter between sensation and perception. From a scene with a wealth of visual stimuli, our attention system can help to block the redundancies so that only the most conspicuous subsets can enter the higher brain regions for complex analysis. Such conspicuous visual subsets, namely, the *salient* targets, are crucial for the human to understand the visual scene. Consequently, the performance of computer applications such as video compression [1], [2], seam carving [3], scene understanding [4], and image retrieval [5]–[7] can be greatly improved by detecting the salient targets in images and videos.

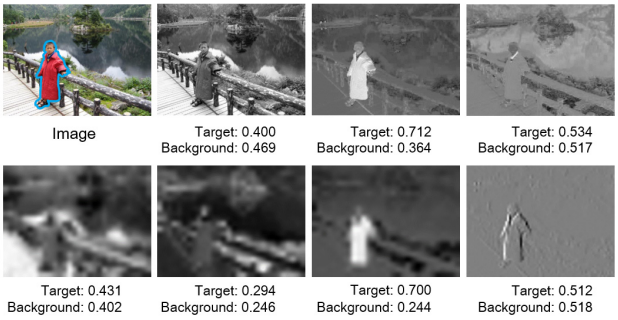

Fig. 1. Different subspaces have different capabilities in distinguishing targets and distractors. The salient target in the first image is marked with a blue contour, while the remaining show the average responses in the target and background region when projecting the image onto different subspaces (e.g., intensity, red/green, and blue/yellow color opponencies; independent components; or sparse basis learned by [15] and [16]). Note that the projected coefficients are normalized to the dynamic range of [0, 1].

To detect salient targets in images, existing studies often compute a saliency map to indicate the important values of all visual subsets. In different approaches, such subsets can be selected as pixels [8], macroblocks [9], or regions [10]. In the computation, a frequently used solution is to represent each subset with multiple visual features and then measure its saliency by fusing the irregularities from various feature channels. For example, Itti *et al.* [9] combined the multiscale center–surround contrasts from intensity, color opponencies, and orientations to generate a final saliency map. Hu *et al.* [11] first turned images into polar space to measure pixel irregularities. Duan *et al.* [12] and Riche *et al.* [13] adopted the YCbCr color space for visual saliency estimation. Goferman *et al.* [14] exploited the influence of high-level features (e.g., the face) for estimating image saliency. For these approaches, a latent hypothesis is that the targets can always be separated from distractors in some specific feature channels. However, the distractors, which may share some visual attributes with salient targets, would be also assigned high saliency scores (e.g., the intensity channel in Fig. 1), leading to a fuzzy saliency map. In fact, an image can be represented by various feature channels, which have different capabilities to distinguish

Manuscript received December 11, 2014; revised October 2, 2015; accepted January 21, 2016. Date of publication February 18, 2016; date of current version May 17, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61370113, Grant 61532003, Grant 61390515, and Grant 61425025, in part by the National Key Technology Research and Development Program under Grant 2014BAK10B02, and in part by the Fundamental Research Funds for the Central Universities. *(Corresponding authors: Jia Li and Yonghong Tian.)*

S. Fang, Y. Tian, and T. Huang are with the School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China, and also with the Cooperative Medianet Innovation Center, China (e-mail: sfang@pku.edu.cn; yhtian@pku.edu.cn; tjhuang@pku.edu.cn).

J. Li is with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing 100191, China, and also with the International Research Institute for Multidisciplinary Science, Beihang University, Beijing 100191, China (e-mail: jiali@buaa.edu.cn).

X. Chen is with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing 100191, China (e-mail: chen@buaa.edu.cn).

targets and distractors (as shown in Fig. 1). Without loss of generality, we represent each channel as a *subspace*, and the objective of saliency estimation can be formulated as finding the optimal subspaces that perform the best in popping out targets and suppressing distractors.

To address this problem, two solutions have been proposed in recent studies. In the first solution, approaches such as [15]–[17] were proposed to obtain a set of subspaces through unsupervised learning [e.g., independent component analysis (ICA) and sparse coding theory]. In these approaches, a set of sparse codes (or sparse functions or visual words) were first learned from massive patches sampled from training images. Images are then projected onto subspaces formed by these sparse codes for saliency estimation. In this manner, redundancies in the input visual stimuli can be greatly removed, and it often becomes easier to detect salient regions in a set of independent subspaces. Instead of learning new subspaces, approaches in the second solution aimed to seek an optimal combination of the classical feature subspaces such as local contrasts [18]; low-, mid-, and high-level features [19]; and wavelet energy [20]. In these approaches, the supervised learning algorithms are often adopted to train the best feature-saliency mapping model on training images with user annotations (e.g., fixations or binary masks of salient objects). In general, all these approaches can achieve impressive performance but also have some drawbacks. The approaches in the first solution may pop out distractors mistakenly due to the heuristic integration of saliency from various independent subspaces, while the approaches in the second solution may not work if all the predefined subspaces fail to distinguish targets from distractors. Therefore, it is necessary to integrate advantages of the two solutions by building candidate subspaces from unsupervised learning and then optimally integrate them from supervised learning.

Inspired by this idea, we propose a novel approach to learn discriminative subspaces for image saliency estimation. In our approach, we first conduct principal component analysis (PCA) on massive randomly selected natural image patches. The principal components, which correspond to the largest eigenvalues, are then selected to build candidate subspaces since they often demonstrate impressive discriminative abilities on targets and distractors. By projecting images on various subspaces, we further characterize each image patch by its average contrasts between randomly selected neighboring and peripheral regions. This random contrast map (RCM) can ensure that the probable target regions have the highest responses, while the lowest responses are assigned to the background. Based on such random contrasts, an optimization framework with pairwise binary terms is adopted to optimally integrate various subspaces under the principle of maximizing the responses of targets while minimizing the responses of distractors so as to separate them perfectly. Experimental results of two public benchmarks show that the proposed approach outperforms 16 approaches (see [15], [18], [21]–[23]) in human fixation prediction and achieves impressive results when processing images with cluttered background and small/medium/large salient objects.

Our main contributions are summarized as follows.

1) We propose to build candidate subspaces from PCA, which can reflect the distribution of the input visual stimuli and perform much better in distinguishing targets from distractors than predefined subspaces.
2) We present an algorithm to characterize an image patch in each subspace with its contrasts against randomly selected neighboring and peripheral regions. Such contrast can be computed with high efficiency to ensure that the probable targets are assigned with the highest responses in most cases.
3) An optimization framework with pairwise binary terms is proposed to learn the most discriminative subspaces for image saliency analysis. The effectiveness of the proposed approach is confirmed by comparisons with other approaches in extensive experiments.

The rest of this paper is organized as follows. Section II reviews related works, and Section III formulates the problem of saliency estimation from the perspective of separating targets and distractors. In Section IV, we describe the details of the proposed approach. Experimental results are presented in Section V, and the entire paper is concluded in Section VI.

II. Related Work

In general, existing approaches on visual saliency estimation mainly differ in two aspects: which subspaces (e.g., color channels; scales; low-, mid-, and high-level features; independent components; and sparse codes) are used and how to integrate the results from various subspaces. According to those differences, most existing saliency models can be categorized into three groups.

Saliency models in the first group adopt predefined subspaces and integrate saliency cues from various subspaces in a heuristic manner. The work of Itti *et al.* [9], which was derived from the idea of [24], was one of the most representative saliency models in the literature. It first represented image in predefined color subspaces such as intensity, red–green, and blue–yellow color opponencies as well as four orientations. After that, local center–surround differences were computed in each subspace to simulate the receptive fields of various neurons. Finally, such local contrasts were integrated with equal weights to produce the final saliency map. Riche *et al.* [21] measured multiscale rarities in color and orientation feature channels and outputted saliency map by intra- and interchannel fusion strategies. Erdem and Erdem [25] directly calculated the distance between the covariance of color, orientation, and spatial features extracted at a local image patch and those of the surrounding patches as its saliency. Gao *et al.* [26] proposed to integrate center–surround differences deviated from intensity, color, and orientation subspaces using the decision theory. Beyond the local cues, some models [13], [27], [28] measured image saliency from the perspective of global rarity. In [22], saliency was derived from the random walking process on a fully connected graph, whose nodes represented image patches and edges were weighted by the mutual similarities. For an image, patches distinctive from the others will be less visited in the random walk and thus become salient. Moreover, some approaches were

proposed to segment images into regions or superpixels. In this manner, the saliency value of a region can be computed by fusing weighted regional contrasts [10], [29]. Sun *et al.* [30] calculated saliency in RGB subspace by analyzing the self-information of the super Gaussian components computed from image patches. Beyond defining saliency as visual rarity, Zhang and Sclaroff [31] computed image saliency by averaging the Boolean maps obtained from lab color channels with random thresholds based on the gestalt principle.

As an extension of spatial feature subspaces, some models were proposed to detect saliency by incorporating the cues from temporal and semantic subspaces. For example, Li *et al.* [32] obtained video saliency by computing regional dissimilarities from motion, color, and texture features. The probabilistic model in [33] integrated the bottom–up saliency map with the learned scene/context priors to produce image saliency. Cerf *et al.* [34] demonstrated that incorporating the face detection result with the bottom–up saliency map of [9] and [22] could achieve a better performance in saliency prediction. Moreover, some models adopted transformed domains for saliency analysis. For example, Hou and Zhang [35] proposed to estimate saliency by analyzing the spectral residual in the amplitude spectrum of image intensity, while Guo *et al.* [36], [37], and Li *et al.* [38] performed the quaternion Fourier transform on multiple feature channels (e.g., intensity, color, and motion) for saliency analysis. Garcia-Diaz *et al.* [39] estimated the optical variability with intensity, spectral wavelengths, and spatial frequency for saliency calculation.

In general, models with predefined subspaces can effectively estimate image/video saliency once targets and distractors can be well separated in certain subspaces. However, those models may fail if all the predefined subspaces cannot separate targets and distractors. Moreover, the heuristic combination of the results from various subspaces often performs unsatisfactory in distractor inhibition. In this case, the estimated saliency map often contains rich noises. As a result, models in the second and the third group proposed to obtain better candidate subspaces and subspace fusion strategies through unsupervised and supervised learning, respectively.

To address the first problem of using predefined subspaces, models in the second group try to learn the *candidate subspaces* from image statistics and *heuristically* fuse saliency cues from these subspaces. Since independent components can decorrelate the input stimuli and, more importantly, they can simulate the receptive fields of simple cells in V1, models [40]–[44] were proposed to learn independent components from massive image patches for the subspace construction. Among these approaches, Bruce and Tsotsos [15] and Hou and Zhang [16] represented image patches by projecting their RGB values onto the learned independent component subspaces. After that, visual saliency was computed from the perspective of self-information or coding length increment. Borji and Itti [17] applied ICA in both RGB and lab color spaces to build candidate subspaces, in which the local and global irregularities were detected and then integrated with several predefined fusion schemas such as $\{+, *, \max, \text{or } \min\}$.

Instead of using independent components, some models [28], [45] adopted principal components in the subspace construction, since the projection onto the principal components with the highest eigenvalues has higher variances. In this manner, fixated and nonfixated locations have steeper contrasts, making it easier to distinguish them [46]. Beyond directly constructing subspaces with ICA and PCA, some approaches were also proposed to learn a set of visual words and their statistical information for saliency computation. For example, Parikh *et al.* [47] calculated saliency based on the co-occurrence of visual words and their spatial information. Li *et al.* [23] proposed to modulate the bottom–up saliency map with foreground and correlation priors between visual words learned from millions of images. In the general case, approaches in the second group can construct better subspaces in a data-driven manner. However, the heuristic combination of subspaces may still lead to noisy saliency maps. Therefore, it is necessary to conduct a certain selection on these learned subspaces to find the most discriminative ones.

To address the second problem of heuristic fusion, models in the third group focus on learning the *optimal fusion strategies* for combining results from various *predefined subspaces*. That is, they conduct some kinds of selections, or namely, reweighting of predefined subspaces. Usually, various kinds of machine learning algorithms are used in these approaches to learn the optimal weights of features from training images with user annotations.

Among these approaches, support vector machine (SVM) is one of the most frequently used learning algorithms. For example, Kienzle *et al.* [48] proposed a nonparametric method to learn the stimulus-saliency mapping function directly from gray image patches by SVM. Judd *et al.* [19] used SVM to train a linear model with the weights of predefined low-level (e.g., the features used in [14] and [15], and so on), mid-level (e.g., the horizon line), and high-level (e.g., the face and the person) features as well as the center prior. Instead of using SVM, Zhao and Koch [18] and Borji [51] adopted Adaboost algorithm for training the saliency model, while Li *et al.* [52] adopted a learning-to-rank framework to learn a linear model for combining the local contrasts of intensity, color, and orientation. Typically, models in this group can achieve impressive performance by selecting and emphasizing the most discriminative subspaces. However, the learned model may fail when none of the predefined subspaces can separate targets from distractors.

To sum up, it is necessary to build effective subspaces that can reflect the distribution of the input stimuli; meanwhile the selection of such subspaces is also an essential step to pop out targets and inhibit distractors. Toward this end, some models were proposed to simultaneously learn the optimal subspaces as well as their fusion strategies in an unified framework. For example, Kanan *et al.* [42] calculated saliency as the probabilistic classifier $log\ p(C = 1|F)$ learned by SVM and independent component features. Yang and Yang [53] computed image saliency by simultaneously training the conditional random field and the visual dictionary. Vig *et al.* [54] searched optimal multilayer predefined feature models and

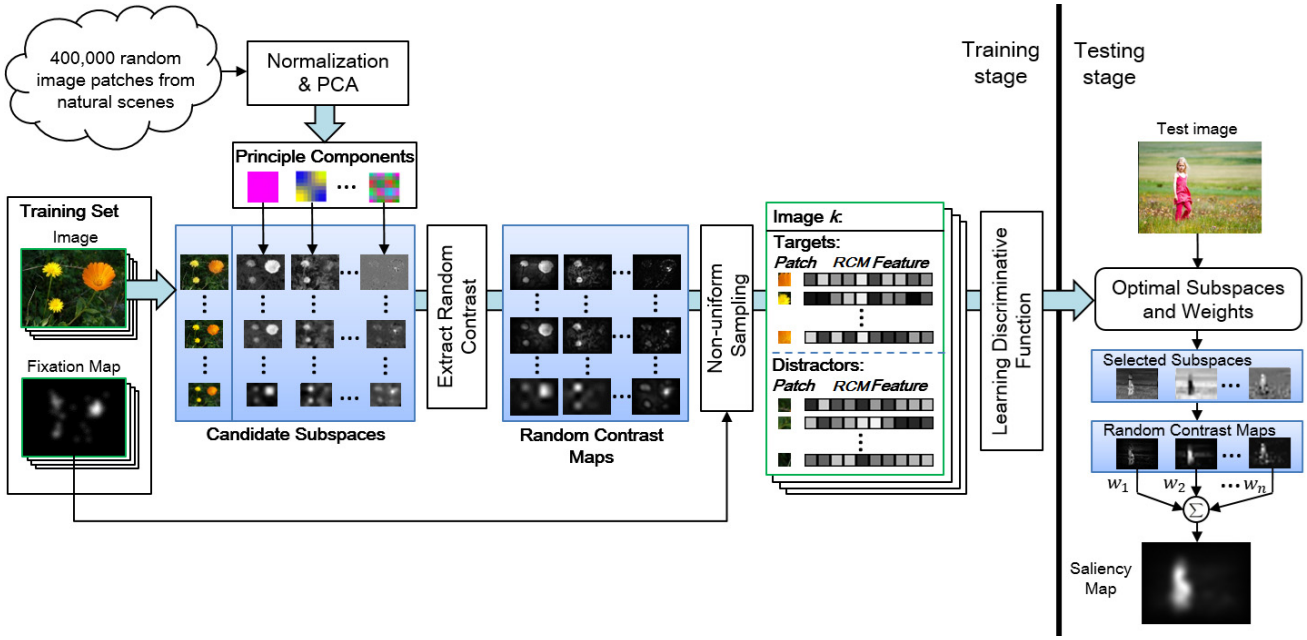

Fig. 2. System framework with a training stage and a testing stage. In the training stage, we first learn a group of principal components from massive image patches randomly sampled from indoor and outdoor scenes, and these principal components are used to build candidate subspaces at multiple scales. In each subspace, we extract an RCM for each training image, and the responses at various RCMs are used to characterize the target and distractor patches nonuniformly sampled from all training images. On these training instances, a saliency model is trained by learning the most discriminative subspaces that perform best in separating targets from distractors within the same image. The learned saliency model, which comprises the optimal subspaces (i.e., principal components and scales) and their combination weights, can be used in the testing stage to efficiently highlight targets and suppress distractors.

learned the optimal combination by SVM. In this paper, we will further explore how to construct effective subspaces through unsupervised learning and select the most discriminative subspaces through supervised learning.

III. Problem Statement

In our approach, we aim to learn a set of subspaces as well as their combination strategies that perform the best in distinguishing targets from distractors. Suppose that we have K training images in total, denoted by $\{I_k\}_{k=1}^K$. Let $\mathbb{T}_k$ and $\mathbb{D}_k$ be the sets of targets and distractors in I_k, respectively. For a target $\mathcal{T} \in \mathbb{T}_k$ (or a distractor $\mathcal{D} \in \mathbb{D}_k$), we can represent it with $\phi_n(\mathcal{T}) \in \mathbb{R}$ [or $\phi_n(\mathcal{D}) \in \mathbb{R}$] in the nth subspace, which is denoted by $\mathcal{S}_n$. Without loss of generality, we assume that there are totally N candidate subspaces and characterize $\mathcal{T}$ (or $\mathcal{D}$) by a column vector $\phi(\mathcal{T})$ [or $\phi(\mathcal{D})$]

$$\phi(\mathcal{T}) = (\phi_1(\mathcal{T}), \ldots, \phi_N(\mathcal{T}))^T. \tag{1}$$

Consequently, the objective of learning the visual saliency model can be described as optimizing a feature-saliency mapping model $F(\cdot)$ that performs the best in distinguishing targets and distractors

$$\begin{aligned} \min_{\pi} \sum_{k=1}^{K} \sum_{\mathcal{T}\in\mathbb{T}_k} \sum_{\mathcal{D}\in\mathbb{D}_k} \delta(F(\pi;\phi(\mathcal{T})) < F(\pi;\phi(\mathcal{D}))) \\ \text{s.t. } 0 \le F(\pi;\phi(\mathcal{T})) \le 1 \quad \forall \mathcal{T} \in \mathbb{T}_k,\ k = 1,\ldots,K \\ 0 \le F(\pi;\phi(\mathcal{D})) \le 1 \quad \forall \mathcal{D} \in \mathbb{D}_k,\ k = 1,\ldots,K \end{aligned} \tag{2}$$

where π is the hyperparameter of $F(\cdot)$. Here, $\delta(\mathbf{e})$ is an indicator function whose value is determined by event $\mathbf{e}$

$$\delta(\mathbf{e}) = \begin{cases} 1, & \text{if } \mathbf{e} \text{ holds} \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

From (2), we can see that $F(\cdot)$ should assign high saliency values to targets and low saliency values to distractors so as to avoid the penalty. Compared with the optimization problem in [18], [19], [48], and [51], the optimization problem in (2) aims to integrate the features from various subspaces that perform the best in distinguishing the targets from the distractors in the same image.

To solve the problem in (2), we can see that there are three subproblems that should be addressed.

1) How to construct a set of subspaces $\{\mathcal{S}_n\}_{n=1}^N$ that can effectively separate targets and distractors.
2) How to find a feature $\phi_n(\cdot)$ to characterize targets and distractors in the nth subspace.
3) How to learn an optimal feature-saliency mapping function that performs the best in distinguishing targets from distractors.

In the next section, these three subproblems will be addressed. After that, we will introduce how to estimate saliency scores with the learned feature-saliency mapping function as well as the features derived from the learned subspaces.

IV. Visual Saliency From Subspace Analysis

The system framework of our model is presented in Fig. 2. From this framework, we can see that three major steps are involved in training the saliency model, including: 1) building

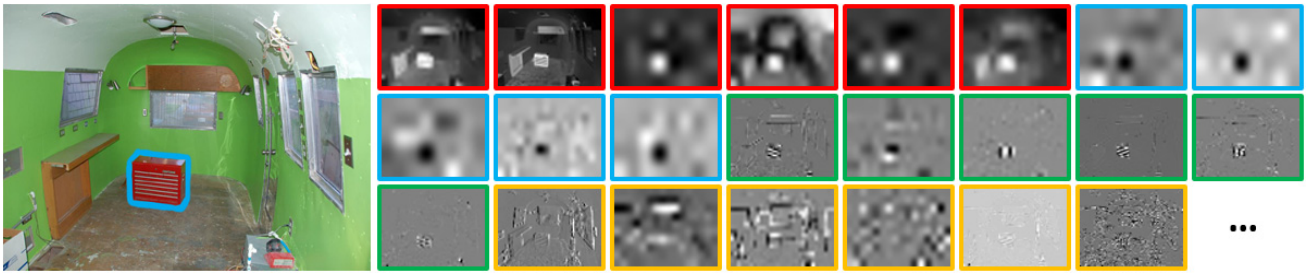

Fig. 3. Representative coefficient map samples from candidate subspaces for one input image. These samples are randomly chosen and labeled with different colors. Red: high responses at targets and low responses at the background. Blue: low responses at targets and high responses at the background. Green: high and low responses at various parts of targets and average responses at the background. Orange: inseparable targets and the background.

candidate subspaces; 2) characterizing image patches in each subspace; and 3) learning the discriminative subspaces and their combination strategies. In the rest of this section, we will focus on introducing the technical details of these three steps as well as the way to compute saliency with the learned saliency model.

A. Building Candidate Subspaces

When projecting an image onto a given subspace, each image patch will obtain a projection coefficient. From the perspective of signal processing, such coefficients should demonstrate a large diversity so as to facilitate the detection of irregular patches. Toward this end, we propose to build a set of candidate subspaces by using PCA. First, we collect 1000 natural images from indoor and outdoor scenes and extract 400 000 nonoverlapping patches (8×8 macroblocks in this paper) from these images. For each patch, we extract a color vector with $8 \times 8 \times 3 = 192$ components, which consists of lab colors of all pixels. Note that we normalize all the components in the color vector to have zero mean and unit standard deviation. By using the PCA algorithm on the covariance matrix of the normalized color vectors, we can get $D = 192$ principal components $\{\xi_1, \xi_2, \ldots, \xi_D\}$ that correspond to the eigenvalues $\lambda_1 > \lambda_2 > \cdots > \lambda_D$. Often, the principal components corresponding to larger eigenvalues are believed to have a better capability in separating the conspicuous image content from the background. Thus, we empirically select the first d principal components to build the candidate subspaces (the influence of d will be discussed in Section V).

When building the candidate subspaces, we often expect that they have the capability to handle targets and distractors at various scales. Toward this end, we propose to build N subspaces totally from the selected d principal components, while the nth subspace is denoted by $\mathcal{S}_n = (s_n, \xi_n)$. Here $s_n \in \{0, 1, 2\}$ is the scale factor and $\xi_n \in \{\xi_1, \xi_2, \ldots, \xi_d\}$ is the principal component. To project an image on the subspace $\mathcal{S}_n$, we first convolve it with a Gaussian smoothing kernel $G(\sigma_n)$ ($\sigma_0 = 2.5$, $\sigma_n = (s_n + 1)^{1/2} \cdot \sigma_0$) and the smoothed image is downsampled by a factor of 2^{s_n} (note that the original image is directly used when $s_n = 0$). From the smoothed image, we can extract a set of nonoverlapping 8×8 patches, and the projection coefficient for the patch at the ith row and jth column can be computed from its normalized color vector $\mathbf{p}_{ij}$

$$\varphi_n(\mathbf{p}_{ij}) = \xi_n^T \mathbf{p}_{ij}. \tag{4}$$

Given the N candidate subspaces, we can obtain a set of coefficient maps, denoted by $\{\varphi_n\}_{n=1}^N$, for an input image. In Fig. 3, we demonstrate several representative coefficient maps when projecting an image onto candidate subspaces. We find that there exist four categories of coefficient maps. The maps in the first category (marked with a red border) achieve high responses at targets and low responses at the background, while the maps in the second category (marked with a blue border) act oppositely. Coefficient maps in the third category, which are marked with a green border, achieve high and low responses at different parts of targets, while the background has average responses. Moreover, there also exist some coefficient maps (marked with an orange border) that fail to distinguish targets from distractors. Therefore, it is necessary to seek effective features to further pop out targets and suppress distractors in the coefficient maps from the first three categories, while the subspaces from the fourth category should be removed by a learning algorithm.

B. Computing Random Contrast Map

Before learning the discriminative subspaces, we have to *assign the highest responses to salient targets in the subspaces from the first three categories*. By observing the responses of a large number of coefficient maps in the candidate subspaces, we find three rules that can be effective to discriminate targets from distractors in subspaces.

1) Targets may appear at any position and scale.
2) Distractors often appear around image borders.
3) Targets are visually distinct from its surroundings.

According to these rules, we find that the responses at targets in a specific subspace can be adjusted as the differences between the projection coefficients of targets and its surroundings. Furthermore, the differences between targets and probable distractors could also help to increase the responses at targets. Ideally, to handle salient targets with different sizes, we can randomly select a set of rectangles with different sizes and aspect ratios for each patch (i.e., flexible scales). In this manner, our approach can adapt to both small and large salient objects by using random rectangles. By enumerating the regional contrasts between various center and peripheral

Algorithm 1 *Ad Hoc* Computation of RCM

Input : Coefficient map φ, number of rectangles C.
Output: Random contrast map ϕ
begin
  Set $\phi^{(0)}(\mathbf{p}_{ij}) = 0; \forall i, j;$
  for $c \leftarrow 1$ ***to*** C **do**
    Generate a random rectangle R_c;
    Compute μ_c^{in} and μ_c^{out} using (6);
    for $\forall (i, j) \in R_c$ **do**
      $\phi^{(c)}(\mathbf{p}_{ij}) \leftarrow \phi^{(c-1)}(\mathbf{p}_{ij}) + \min(|\varphi(\mathbf{p}_{ij}) - \mu_c^{in}|, |\varphi(\mathbf{p}_{ij}) - \mu_c^{out}|);$
    end
  end
  $\phi(\mathbf{p}_{ij}) = \phi^{(C)}(\mathbf{p}_{ij})/C, \forall i, j;$
end

regions [55], we can pop out the salient targets at any scale. However, it can be extremely time-consuming to test a large number of rectangles (i.e., probable scales) for each patch to find its probable scale, making this solution infeasible for applications in real scenarios.

Inspired by [56], we randomly select only C rectangles in the nth subspace for each projecting coefficient map, denoted as $\{R_{\text{nc}}\}_{c=1}^{C}$. With these rectangles, the random contrasts for a patch at (i, j), denoted by $\phi_n(\mathbf{p}_{ij})$, is computed as

$$\phi_n(\mathbf{p}_{ij}) = \frac{1}{C}\sum_{c=1}^{C}\delta((i,j) \in R_{\text{nc}}) \cdot \min\left(\left|\varphi_n(\mathbf{p}_{ij}) - \mu_{\text{nc}}^{\text{in}}\right|, \left|\varphi_n(\mathbf{p}_{ij}) - \mu_{\text{nc}}^{\text{out}}\right|\right) \quad (5)$$

where $\mu_{\text{nc}}^{\text{in}}$ and $\mu_{\text{nc}}^{\text{out}}$ are the average responses inside and outside the rectangle R_{nc}

$$\mu_{\text{nc}}^{\text{in}} = \frac{\sum_{(i,j)\in R_{\text{nc}}}\varphi_n(\mathbf{p}_{ij})}{\sum_{(i,j)\in R_{\text{nc}}}1}, \quad \mu_{\text{nc}}^{\text{out}} = \frac{\sum_{(i,j)\notin R_{\text{nc}}}\varphi_n(\mathbf{p}_{ij})}{\sum_{(i,j)\notin R_{\text{nc}}}1}. \quad (6)$$

The algorithm of RCM computation is summarized in Algorithm 1. Algorithm 1 computes the differences between a patch and multiple randomly selected surrounding regions, which can be viewed as the multiscale local contrasts. Also, the differences against randomly selected peripheral regions are also incorporated in (5) to characterize a patch, which correspond to the contrasts against probable backgrounds. To speed up the computation process, we restrain that the area of each random rectangle should be larger than 1% of the image. By computing the contrasts against the patches inside and outside the rectangles, we can effectively pop out the targets at different scales in the subspaces from the first three categories (as shown in Fig. 5).

In the computation of RCM, the algorithm complexity may become a major concern. Actually, the computation of RCM is extremely efficient, since it adopts only C rectangles. For an image with M patches, each rectangle is expected to cover $M/2$ patches on average (if C is very large). From (5) and (6), we see that only $O(MC)$ additions are required in the computation of RCM. In our implementation, it takes only 0.05 seconds to compute the RCM in a subspace for an image with $M = 128 \times 96$ patches and $C = 10\,000$ rectangles.

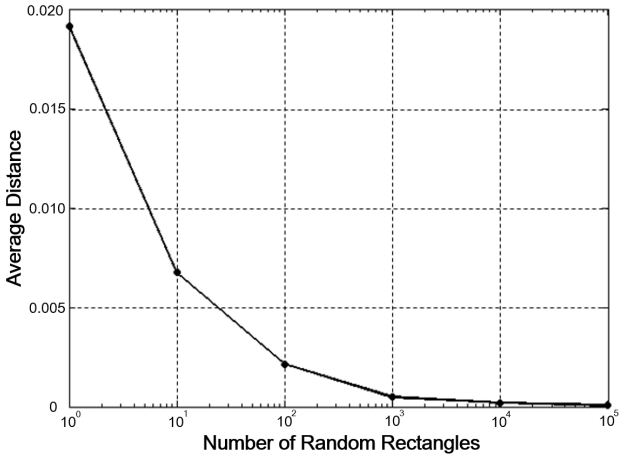

Fig. 4. Stability of using different numbers of random rectangles in RCM computation. In different trails, the average difference between RCMs computed by using the same number of rectangles will become extremely small when sufficient random rectangles are used in the computation.

Moreover, we also conduct an experiment to test the stability of the computed RCMs with Algorithm 1. In the experiment, we collect $Q = 100$ images and project each image to a randomly selected subspace $\mathcal{S}_{n_q}, q = 1, \ldots, Q$. On each of the Q coefficient maps, we compute a set of RCMs in $T = 10$ trails using C rectangles ($C = 1, 10, \ldots, 10^5$). Let $\phi_{qn_q}^{(Cu)}$ and $\phi_{qn_q}^{(Cv)}$ be the RCMs computed in the uth and vth trails with C random rectangles, which are of the qth image in the subspace $\mathcal{S}_{n_q}$, and the difference between them can be computed as the pixelwise $\ell - 2$ distance

$$\text{dist}\left(\phi_{qn_q}^{(Cu)}, \phi_{qn_q}^{(Cv)}\right) = \frac{1}{H_q W_q}\sqrt{\sum_{i=1}^{H_q}\sum_{j=1}^{W_q}\left(\phi_{qn_q}^{(Cu)}(\mathbf{p}_{ij}) - \phi_{qn_q}^{(Cv)}(\mathbf{p}_{ij})\right)^2} \quad (7)$$

where H_q and W_q are the height and width of the projection coefficient map of the qth image in the subspace $\mathcal{S}_{n_q}$, respectively. Given such a pairwise difference, we can further measure the average difference Δ_C between the computed RCMs on the Q images and T trails

$$\Delta_C = \frac{1}{Q(T^2 - T)}\sum_{q=1}^{Q}\sum_{u\neq v}^{T}\text{dist}\left(\phi_{qn_q}^{(Cu)}, \phi_{qn_q}^{(Cv)}\right). \quad (8)$$

As presented in Fig. 4, the average difference between different trails may rapidly converge to an extremely small value with sufficient rectangles. This fact ensures that, given the input image and the subspace, we can obtain almost the same RCMs if sufficient random rectangles are used. Inspired by this idea, we also propose an adaptive algorithm to speed up the computation of RCM. Different from the *ad hoc* algorithm in Algorithm 1, the adaptive algorithm starts from a small number of random rectangles (40 in this paper) and doubles the number of rectangles in each iteration. The iteration, as illustrated in Algorithm 2, will terminate when the computed

Algorithm 2 Adaptive Computation of RCM

Input : Coefficient map φ, threshold ϵ
Output: Random Contrast Map ϕ.
begin
  Set $t = 0$, $C^{(0)} = 40$;
  Compute $\phi^{(0)}$ using Algorithm 1 and $C^{(0)}$;
  repeat
    $t = t + 1$;
    Compute $\widehat{\phi}$ using Algorithm 1 and $C^{(t-1)}$;
    $\phi^{(t)} \leftarrow 0.5\left(\phi^{(t-1)} + \widehat{\phi}\right)$, $C^{(t)} \leftarrow 2C^{(t-1)}$;
  until $dist(\widehat{\phi}, \phi^{(t-1)}) \leq \epsilon$;
  $\phi = \phi^{(t)}$;
end

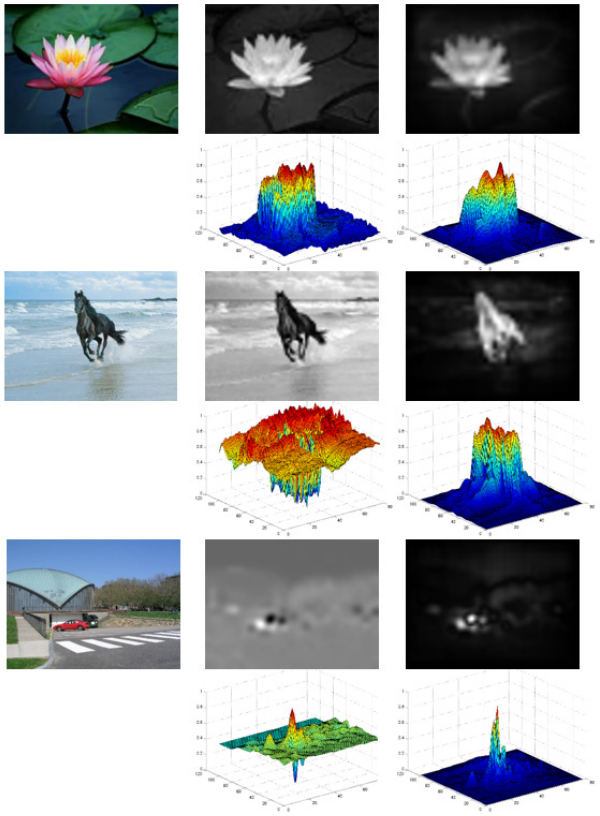

Fig. 5. RCMs can pop out targets in subspaces from the first three categories. For various kinds of coefficient maps (the second column), the RCMs can assign the highest values to targets and the lowest values to distractors so as to separate them (the third column).

RCM converges (i.e., the variation of RCMs in each iteration is smaller than a predefined threshold). In experiments, we find that the algorithm usually converge within three iterations, and the algorithm adopts 122 random rectangles on average when processing each of the 100 testing images.

Some RCMs on the coefficient maps in the first three categories are displayed in Fig. 5. From Fig. 5, we can see that the RCM can effectively assign the highest responses to salient targets with various sizes and lowest responses to distractors in the subspaces from the first three categories.

C. Learning for Subspace Selection and Combination

Typically, RCMs for subspaces from the first three categories can assign high responses to targets at flexible scales and low responses to distractors, while RCMs for subspaces in the last category often fail to separate targets and distractors. Therefore, we have to learn to select and combine RCMs that perform the best in popping out targets and suppressing distractors. Note that RCMs from various subspaces may have different resolutions and dynamic ranges. To facilitate the learning process, we further resize all RCMs to scale 0 and normalize them to the same dynamic range of [0, 1]. Then, we can characterize each 8×8 image patch with

$$\phi(\mathbf{p}_{ij}) = (\phi_1(\mathbf{p}_{ij}), \ldots, \phi_N(\mathbf{p}_{ij}))^T \tag{9}$$

where $\phi(\mathbf{p}_{ij})$ represents the feature of a patch $\mathbf{p}_{ij}$ (i.e., the RCM feature illustrated in Fig. 2).

Given the patch representation, we aim to train a feature-saliency mapping function $0 \leq F(\phi(\mathbf{p}_{ij})) \leq 1$ that performs the best in distinguishing targets from distractors. Actually, $F(\phi(\mathbf{p}_{ij}))$ can take various forms such as linear, exponential, or sigmoid functions. To demonstrate the effectiveness of our approach, we adopt the linear function as follows:

$$F(\phi(\mathbf{p}_{ij})) = \mathbf{w}^T\phi(\mathbf{p}_{ij}), \quad \|\mathbf{w}\|_1 = 1, \quad 0 \preceq \mathbf{w} \preceq 1. \tag{10}$$

Before optimizing the parameter $\mathbf{w}$, we have to select a set of targets and distractors from the training images first. Here we use the sampling strategy proposed in [19] and randomly select ten target patches and ten distractor patches from the top 20% and bottom 70% salient regions on the fixation density maps, respectively. Note that positive training instances are only sampled from the patches with the ground-truth saliency value above 0.05 and the negative ones are selected from the patches with the ground-truth saliency value below 0.05. These patches are then represented by the contrast values at the corresponding locations of RCMs (i.e., random contrasts from various subspaces).

With these training instances, we can optimize $F(\phi(\mathbf{p}_{ij}))$ by solving the binary optimization problem in (2)

$$\begin{aligned} \min_{\mathbf{w}} \; & \sum_{k=1}^{K} \sum_{\mathcal{T} \in \mathbb{T}_k} \sum_{\mathcal{D} \in \mathbb{D}_k} \delta(\mathbf{w}^T\phi(\mathcal{T}) < \mathbf{w}^T\phi(\mathcal{D})) \\ \text{s.t.} \; & \|\mathbf{w}\|_1 = 1 \\ & 0 \preceq \mathbf{w} \preceq 1. \end{aligned} \tag{11}$$

To solve the optimization problem with a set of binary terms, an alternative solution is to minimize its upper bound, instead. Since

$$\delta(\mathbf{w}^T\phi(\mathcal{T}) < \mathbf{w}^T\phi(\mathcal{D})) \leq \exp(\mathbf{w}^T\phi(\mathcal{D}) - \mathbf{w}^T\phi(\mathcal{T})) \tag{12}$$

we can rewrite the optimization problem as

$$\begin{aligned} \min_{\mathbf{w}} \; & \sum_{k=1}^{K} \sum_{\mathcal{T} \in \mathbb{T}_k} \sum_{\mathcal{D} \in \mathbb{D}_k} \exp(\mathbf{w}^T\phi(\mathcal{D}) - \mathbf{w}^T\phi(\mathcal{T})) \\ \text{s.t.} \; & \|\mathbf{w}\|_1 = 1 \\ & 0 \preceq \mathbf{w} \preceq 1. \end{aligned} \tag{13}$$

In this manner, the optimization problem becomes convex with exponential terms and linear constraints only. Therefore, we can use the Lagrangian multiplier method to efficiently solve it and reach the global minimum. The best parameter $\mathbf{w}^*$, which actually assigns a positive weight for each candidate subspace, can be used for saliency prediction of new images. In our experiment, we find that weights of most subspaces are extremely small, which indicates that the contribution of these subspaces, as well as the RCMs, is very low in saliency computation. To speed up the computation process, we only select subspaces with the highest weights that sum up to 0.99, while all the other subspaces are ignored in saliency computation (i.e., their weights are set to zero). For the sake of simplicity, the adjusted parameters are denoted by $\widehat{\mathbf{w}}$.

Given a testing image, we can project it into the selected subspaces and extract a set of RCMs from the projection coefficient maps. These RCMs are then linearly combined to generate the saliency map using the learned weights $\widehat{\mathbf{w}}$. Since the additive linear combination often leads to a fuzzy saliency map, we adopt the normalization step presented in [22] twice. In the normalization, the random walking is performed on the saliency map to converge energy to the most salient locations predicted by our approach, making the background much cleaner. After that, we adopt a Gaussian convolution with $\sigma = 2.5$ to further emphasize the regions around the most salient locations.

V. Experiments

In this section, we aim to validate the effectiveness and analyze the advantage of the proposed approach in image saliency estimation with several experiments. Toward this end, we utilize two classical benchmarks in the experiments.

1) *MIT1003:* This benchmark was provided by [19]. It consists of 1003 images and the corresponding human fixations obtained under free-viewing conditions. In the experiment, we use tenfold cross validation. That is, we divide MIT1003 into ten subsets, each with 100 images (one subset contains 103 images). Each time, we train a model with nine subsets and test it on the rest subset, and we obtain 1003 saliency maps in this manner. Note that the tenfold cross validation is repeated ten times with different random splits of MIT1003, and the mean and standard deviation of performance scores are reported.
2) *ImgSal:* This benchmark was first proposed in [38]. It contains 235 color images, including 50 images with large salient regions, 80 images with intermediate salient regions, 60 images with small salient regions, 15 images with cluttered background, 15 images with repeating distractors, and 15 images with both large and small salient regions. On this benchmark, we conduct a set of experiments to further exploit the advantage of our proposed approach.

In the experiments, 16 state-of-the-art approaches are used for comparison. Beyond EDN [54], we can roughly categorize these approaches into three groups from the perspective of subspace generation and subspace combination.

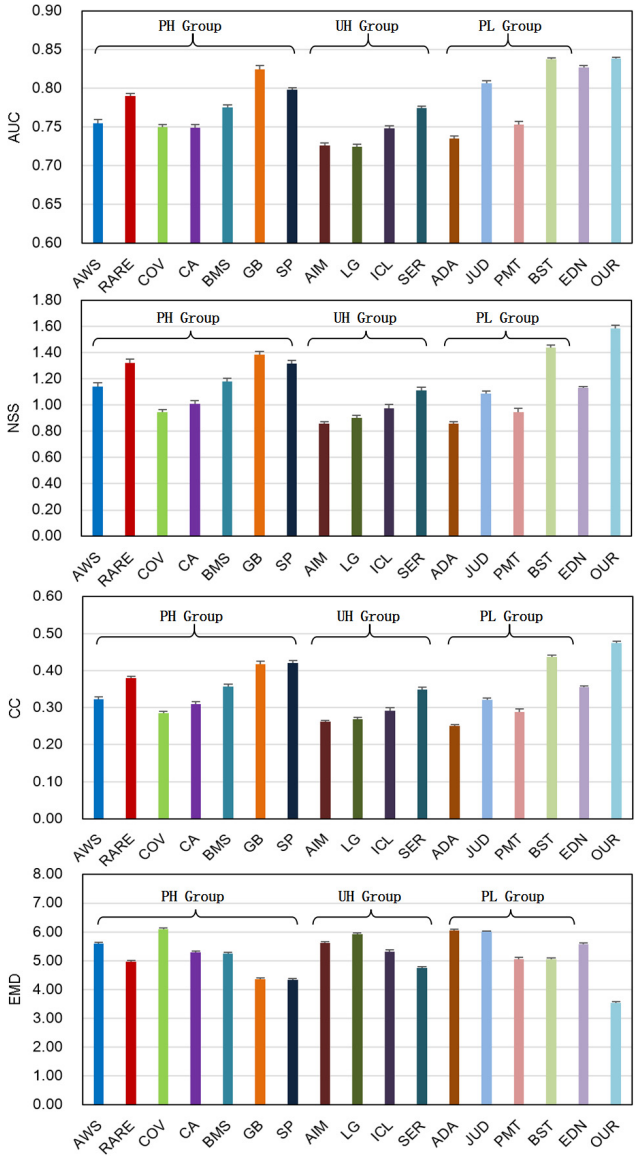

Fig. 6. Performances of 17 models on MIT1003. The error bar is $\sigma/\sqrt{1003}$, where σ is the standard deviation of performance scores. Our approach (denoted by OUR) outperforms all the other 16 models, while the AUC score is comparable with BST (AUC, NSS, and CC: the higher, the better; EMD: the lower, the better).

1) *PH Group:* This group contains seven bottom–up approaches that adopt *predefined* subspaces (i.e., preattentive features) and *heuristic* integration strategies, including AWS [39], GB [22], RARE [21], COV [25], CA [14], BMS [31], and SP [23].
2) *UH Group:* This group contains four approaches that heuristically combine the subspaces learned through unsupervised image statistics (i.e., independent components, sparse codes, etc.). Approaches in this group include AIM [15], ICL [16], SER [41], and LG [17].
3) *PL Group:* This group contains four top–down methods that aim to combine *predefined* subspaces through supervised learning, including JUD [19], BST [51], ADA [18], and PMT [20].

Fig. 7. Sample saliency maps generated by representative eight saliency models on the MIT1003 benchmark.

When compared with these models, we use the area under the ROC curve (AUC),[1] the normalized scanpath saliency (NSS), the linear correlation coefficient (CC), and the earth mover's distance (EMD)[2] for performance evaluation.

In the comparison, the number of scales is set to 3 and all the 192 principal components are used in the training process (i.e., we have $3 \times 192 = 576$ candidate subspaces). We also conduct several experiments to demonstrate the performance (AUC and time cost) when the number of scales and principal components vary.

A. Comparison With the State of the Arts

In the first experiment, we compare our approach with the other 16 approaches on the MIT1003 benchmark. The performance of these approaches are illustrated in Fig. 6 and some representative results are demonstrated in Fig. 7.

As shown in Fig. 6, saliency models in the PH group can achieve promising performances with predefined

[1]Note that there are many ways to compute AUC, and we use the code provided by Judd *et al.* [19], which is available at http://people.csail.mit.edu/tjudd/WherePeopleLook/index.html.

[2]Models that can better predict human fixations exhibit lower EMD and higher AUC, NSS, and CC.

feature subspaces. Surprisingly, some models in the UH group, which can learn a large number of subspaces from image statistics, may perform even worse than some models in the PH group (e.g., CA outperforms AIM and LG). This indicates that beyond the generation of subspaces, the selection of subspaces is also an essential step in building a saliency model since we have demonstrated that targets and distractors may become inseparable in some subspaces. As shown in Fig. 7, the saliency maps may become very fuzzy by directly incorporating the results from such predefined subspaces.

The models in PL group, which focus on the selection of predefined subspaces, achieve impressive performances and outperform most of the models in PH group and UH group. This implies that the selection of subspaces is much more important than the generation of candidate subspaces. However, the limited number of predefined subspaces may reduce the performance of these learning-based models. Actually, more candidate subspaces often facilitate performances, even with the same learning algorithm. For example, both BST and ADA adopt the boosting algorithm, while BST utilizes much more candidate subspaces. In this manner, BST obtains better performances (AUC $= 0.837$, NSS $= 1.438$, CC $= 0.437$, and EMD $= 5.055$) than ADA (AUC $= 0.735$, NSS $= 0.857$, CC $= 0.250$, and EMD $= 6.053$).

From Fig. 6 and the scores on all images of MIT1003, we find that the AUC, NSS, CC, and EMD scores of our approach are significantly higher (paired t-test, $p < 0.05$) than those of the other 16 approaches (comparable only with BST in terms of AUC). On the one hand, our approach learns the candidate subspaces through PCA, which can characterize the distribution of input visual stimuli. By incorporating the scale factor, candidate subspaces could handle targets and distractors with different sizes. In particular, once a subspace can separate targets and distractors, the RCMs can assign the highest responses to targets with various sizes and the lowest responses to distractors, which may facilitate the learning process. On the other hand, our learning algorithm aims to separate a set of target–distractor pairs, which, as a consequence, can select the subspaces that perform the best in separating targets and distractors in each image. Compared with the classification framework used in models from the PL group and EDN, we just separate targets and distractors to avoid directly classifying patches as targets or distractors. In this manner, our linear model performs better (AUC $= 0.837$, NSS $= 1.584$, CC $= 0.474$, and EMD $= 3.544$) than the models learned in ADA, JUD, PMT, and BST. Moreover, JUD and BST also utilize various mid-level (e.g., the horizon line) and high-level features (e.g., the face and the person), which may prevent the usage of these approaches in general scenarios. For example, saliency map of the fifth image in Fig. 7 produced by the JUD model could accurately highlight the walkers by using the person detector, but such detector is far from perfect and may bring in false alarms in real-world scenes. On the contrary, we only use low-level features in our approach, implying that our approach can adapt to various application scenarios in popping out targets and suppressing distractors.

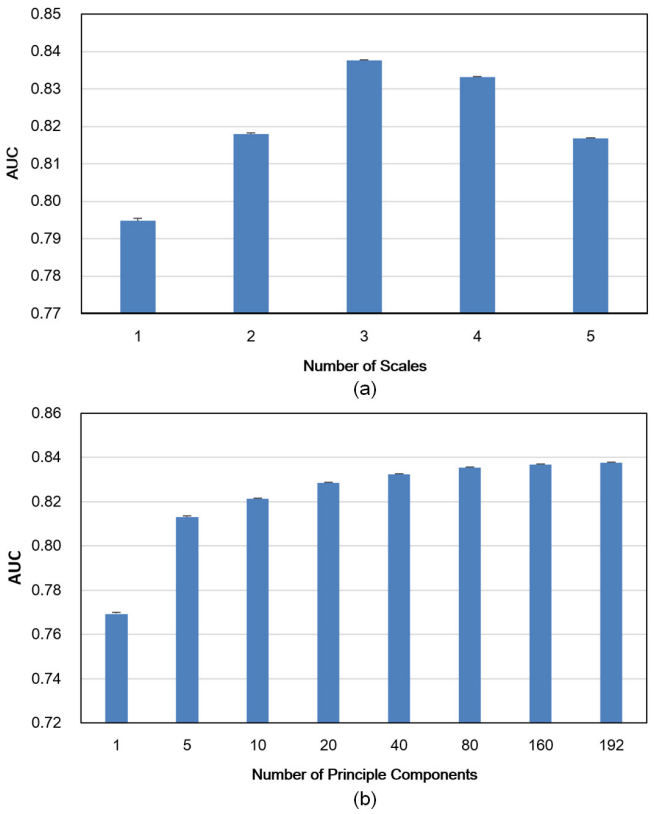

Fig. 8. Performance of our approach when using different numbers of scales and principal components. We can see that our approach performs the best when using three scales and all principal components. (a) Performance with different numbers of scales; (b) Performance with different numbers of principal components.

B. Performance Analysis

Beyond the comparisons with the state of the art, we also conduct several experiments to show the influence of parameters as well as the computational complexity. In the first experiment, we vary the number of scales and the number of principal components to check the performance on MIT1003 benchmark. The experimental results are shown in Fig. 8.

From Fig. 8(a), we see that our approach performs the best when using three scales (i.e., the original and its 1/2 and 1/4 versions). These scales are often sufficient to cover the salient targets with various sizes. When using four or five scales, the input image downsampled to a very small resolution, and an image patch may simultaneously cover parts of the target as well as the background region in most cases, leading to inaccurate RCMs. We also find from Fig. 8(b) that our approach performs the best when using all the $d = 192$ principal components. In particular, using the first several principal components corresponding to the largest eigenvalues can already achieve the impressive performance, while incorporating more principal components can gradually improve the overall performance. Generally speaking, more principal components (even the ones corresponding to small eigenvalues) can provide us more information about the input image, while our learning algorithm can select the most useful information (i.e., subspaces formed by the best principal components at the best scales) for saliency prediction.

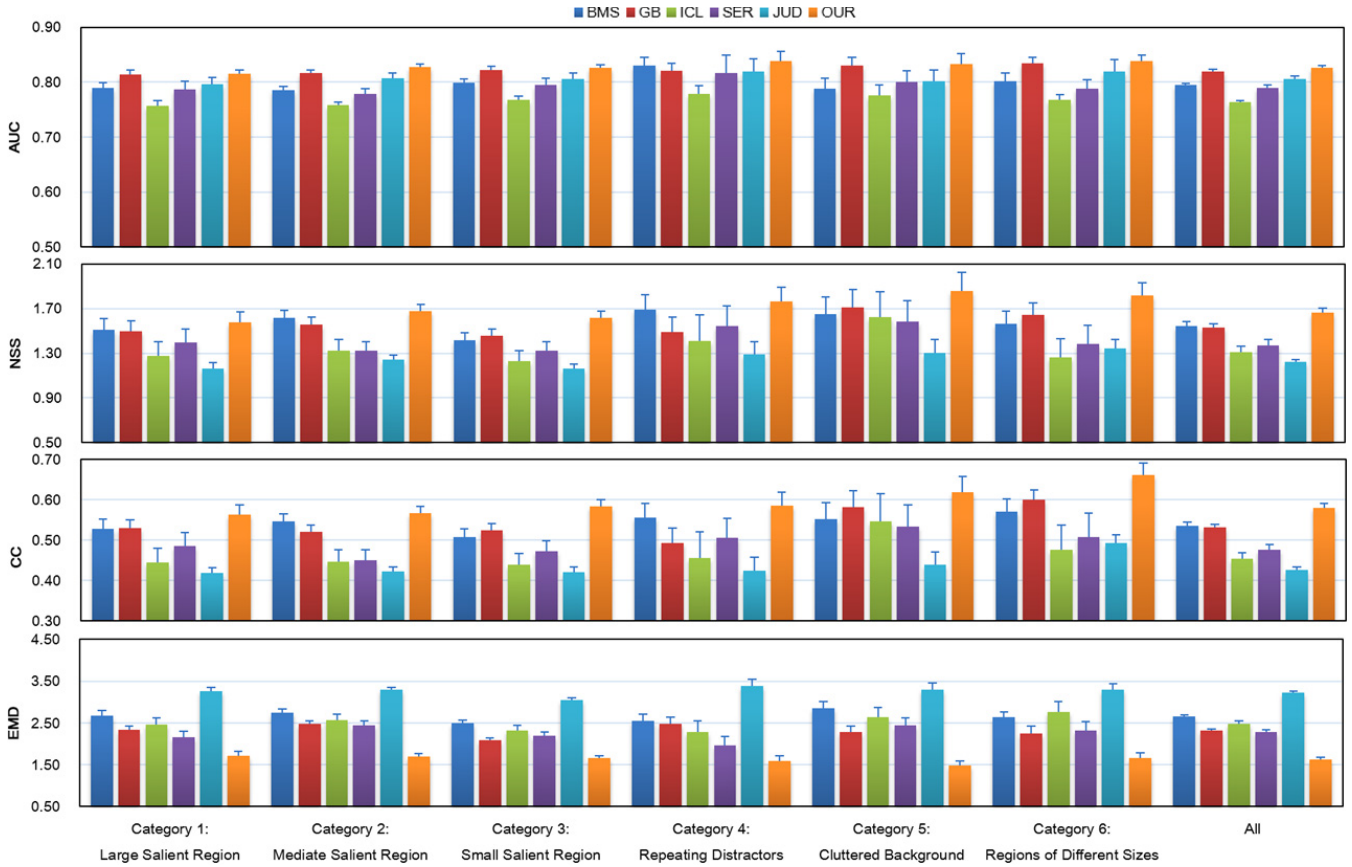

Fig. 9. Model comparison of six models on the ImgSal data set. Performances on every single category and the whole benchmark are displayed.

Fig. 10. Representative saliency maps generated by our approach on the ImgSal data set.

From the result of the first experiment, two major concerns may arise: what is the performance of the proposed approach when handling images that contain objects with different sizes and how fast is the approach when using 192 principal components?

To demonstrate the performance of our approach in processing objects with different sizes, we conduct an experiment on the ImgSal benchmark. ImgSal groups 235 images into six categories according to the object size and background complexity. In particular, we adopt the model trained on all images from MIT1003 benchmark to demonstrate its generalization ability. The performance of our approach and some other saliency models that perform among the best on the MIT1003 data set (i.e., GB and BMS from the PH group, ICL, and SER from the UH group, JUD from the PL group), are demonstrated in Fig. 9. From Fig. 9, we can see that our saliency model performs the best in all the six categories. Note that our saliency model is trained on the MIT1003 benchmark, which indicates that we incorporate no data set bias in the saliency model. From the representative saliency maps in Fig. 10, our approach demonstrates impressive performance even for large salient objects without any presegmentation. Actually, the proposed RCMs operate on flexible scales. In this manner, salient targets with various sizes can be highlighted by using flexible center and peripheral regions defined by random rectangles. Even when the background becomes very complex (e.g., category 4 and category 5), our model can still perfectly pop out the salient target by selecting the most effective subspaces for distractor inhibition. To sum up, the experimental results on these six categories prove the generalization ability of our approach as well as its effectiveness to handle various scenarios.

To further quantize the performance of our approach in processing images with obvious salient objects, we test

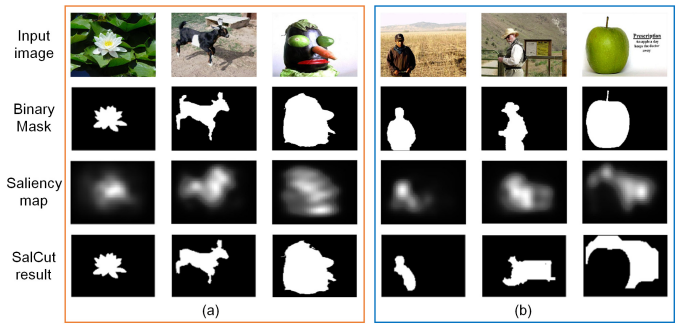

Fig. 11. Representative results of our approach on MSRA10K. (a) Successful examples. (b) Failures.

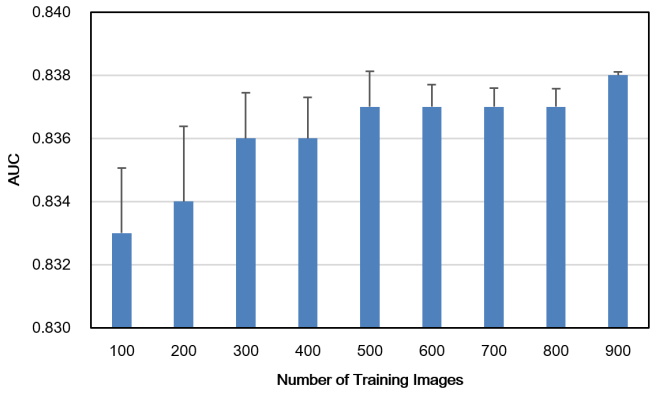

Fig. 12. Mean and standard deviation of AUC scores of our approach on MIT1003 with models trained on different numbers of training images.

our approach on a large salient object detection data set MSRA10K. We first split images into superpixels, and the saliency of each superpixel is set to the average saliency values of the patches it contains. After that, SalCut algorithm [57] is adopted to segment the accurate boundaries of salient objects. With such a simple postprocessing, the F-Measure score of our approach reaches 0.838, which is comparable with GMR [58] (0.839), DSR [59] (0.833), and HC [10] (0.740), but worse than RC [60] (0.875) and DRFI [61] (0.905). To explain the results, we show some representative results, including failure examples, in Fig. 11. From Fig. 11, we can see that our approach can successfully highlight the small, medium, and large salient objects when the salient object can pop out as a whole [see Fig. 11(a)]. However, the proposed framework, which is designed for fixation prediction, tends to pop out only the most salient locations in many cases [see Fig. 11(b)]. In these cases, it is difficult to pop out the whole salient objects, even though human fixations are successfully predicted. From these results, we can see that our approach can facilitate the segmentation of the most salient objects to some extent, since it can successfully locate the most salient locations.

Beyond the performance on saliency prediction, we also conduct the third experiment to address the concern on computational complexity. Note that the main step in predicting saliency of a testing image is the feature extraction, which will be the input of a linear model for saliency map computation.

TABLE I
TIME COST IN OUR FEATURE EXTRACTION STEPS

# candidate subspaces	# selected subspaces	time cost (s)
3	2	0.019
15	6	0.037
30	8	0.041
60	10	0.042
120	14	0.048
240	17	0.054
480	17	0.053
576	17	0.052

Therefore, we only test the speed of feature extraction when using different numbers of principal components. Intuitively, more principal components will generate more candidate subspaces, making the time cost in the training process grow linearly with the number of principal components. However, after the training stage, only limited number of candidate subspaces are selected while the others are discarded, making the testing process extremely efficient.

To validate this, the third experiment consists of two parts: 1) how many subspaces are selected given different numbers of candidate subspaces and 2) how long it will take on average to extract features with the selected subspaces. The results are presented in Table I. We can see that our algorithm actually selects only the most discriminative 17 subspaces (i.e., only 3%), even using 576 candidate subspaces (i.e., three scales and 192 principal components). In the testing process, only the RCMs in these 17 subspaces are computed with the adaptive algorithm in Algorithm 2, achieving an efficient testing process. With a 3.2-GHz CPU, our C++ implementation takes 0.052 s on average to extract all the required features for each of the 235 testing images with a resolution of 640×480. This indicates that our approach can be used even to perform the nearly real-time analysis for live video streams (e.g., 640×480 surveillance video).

Finally, we conduct the last experiment to validate the influence of the number of training images. The objective is to check whether 900 training images are sufficient to train the model and whether the overall performance can be further improved by incorporating more training data. In the experiment, tenfold cross-validation is used as well. Each time, training images are first randomly selected from the nine training subsets (i.e., 90% training images), and the learned model is tested on the testing subset (i.e., 10% testing images). By repeating the training and testing processes for ten times, we generate 1003 saliency maps as well. The experimental results are shown in Fig. 12, from which we can see that even with 100 training images, the performance already becomes very impressive. This indicates that our approach can effectively select the most discriminative subspaces by focusing on separating targets and distractors. At the same time, by bringing in more training images, the risk of overfitting becomes lower, leading to higher AUCs. As the consequence, we can safely assume that incorporating more training images can improve performances. When the number of training images grows from 100 to 900, the AUC score converges to 0.838,

which implies that 900 images are sufficient to train a robust saliency model that can be generalized to various scenarios.

VI. Conclusion

This paper presents an approach that learns discriminative subspaces for image saliency estimation. The subspaces learned from PCA can well reveal the distribution of the input stimuli. Furthermore, we propose to compute the RCMs from the projecting coefficient maps on these candidate subspaces to ensure that targets with various sizes have the highest responses while the distractors' responses are greatly suppressed. In the learning algorithm, we focus on separating targets and distractors instead of seeking a classifier that directly recognizes targets and distractors. Experimental results show that this mechanism is very effective in selecting a small number of discriminative subspaces out of a large number of candidates, which ensures the reliability and efficiency of the proposed approach.

In the future work, we will try to expand the generation of candidate subspaces by incorporating semantic subspaces as well as the spectral information. We will also incorporate our fixation prediction result with superpixel segmentations to generate finer saliency maps. Moreover, we will exploit various kinds of prior knowledge in subspace selection, which can be derived from tasks, spatiotemporal information or massive image statistics.

References

[1] L. Itti, "Automatic foveation for video compression using a neurobiological model of visual attention," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1304–1318, Oct. 2004.

[2] Z. Li, S. Qin, and L. Itti, "Visual attention guided bit allocation in video compression," *Image Vis. Comput.*, vol. 29, no. 1, pp. 1–14, Jan. 2011.

[3] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," in *Proc. ACM SIGGRAPH*, New York, NY, USA, 2007, pp. 1–10.

[4] C. Siagian and L. Itti, "Rapid biologically-inspired scene classification using features shared with visual attention," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 2, pp. 300–312, Feb. 2007.

[5] H. Fu, Z. Chi, and D. Feng, "Attention-driven image interpretation with application to image retrieval," *Pattern Recognit.*, vol. 39, no. 9, pp. 1604–1621, Sep. 2006.

[6] S. Wei, Y. Zhao, C. Zhu, C. Xu, and Z. Zhu, "Frame fusion for video copy detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 1, pp. 15–28, Jan. 2011.

[7] S. Wei, D. Xu, X. Li, and Y. Zhao, "Joint optimization toward effective and efficient image search," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 2216–2227, Dec. 2013.

[8] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 1597–1604.

[9] 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.

[10] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 409–416.

[11] Y. Hu, D. Rajan, and L.-T. Chia, "Adaptive local context suppression of multiple cues for salient visual attention detection," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2005, pp. 1–4.

[12] L. Duan, C. Wu, J. Miao, L. Qing, and Y. Fu, "Visual saliency detection by spatially weighted dissimilarity," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 473–480.

[13] N. Riche, M. Mancas, B. Gosselin, and T. Dutoit, "Rare: A new bottom-up saliency model," in *Proc. IEEE Conf. Image Process. (ICIP)*, Sep./Oct. 2012, pp. 641–644.

[14] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 2376–2383.

[15] N. D. Bruce and J. K. Tsotsos, "Saliency based on information maximization," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Vancouver, BC, Canada, 2005, pp. 155–162.

[16] X. Hou and L. Zhang, "Dynamic visual attention: Searching for coding length increments," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2009, pp. 681–688.

[17] A. Borji and L. Itti, "Exploiting local and global patch rarities for saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 478–485.

[18] Q. Zhao and C. Koch, "Learning visual saliency by combining feature maps in a nonlinear manner using AdaBoost," *J. Vis.*, vol. 12, no. 6, p. 22, 2012.

[19] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Sep./Oct. 2009, pp. 2106–2113.

[20] J. Li, Y. Tian, T. Huang, and W. Gao, "Probabilistic multi-task learning for visual saliency estimation in video," *Int. J. Comput. Vis.*, vol. 90, no. 2, pp. 150–165, Nov. 2010.

[21] N. Riche, M. Mancas, M. Duvinage, M. Mibulumukini, B. Gosselin, and T. Dutoit, "RARE2012: A multi-scale rarity-based saliency detection with its comparative statistical analysis," *Signal Process., Image Commun.*, vol. 28, no. 6, pp. 642–658, Jul. 2013.

[22] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2007, pp. 545–552.

[23] J. Li, Y. Tian, and T. Huang, "Visual saliency with statistical priors," *Int. J. Comput. Vis.*, vol. 107, no. 3, pp. 239–253, May 2014.

[24] C. Koch and S. Ullman, "Shifts in selective visual attention: Towards the underlying neural circuitry," in *Matters of Intelligence: Conceptual Structures in Cognitive Neuroscience*. Dordrecht, The Netherlands: Springer, 1987, pp. 115–141.

[25] E. Erdem and A. Erdem, "Visual saliency estimation by nonlinearly integrating features using region covariances," *J. Vis.*, vol. 13, no. 4, p. 11–20, 2013.

[26] D. Gao, V. Mahadevan, and N. Vasconcelos, "The discriminant center-surround hypothesis for bottom-up saliency," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2009, pp. 497–504.

[27] M. A. Kunar, S. J. Flusberg, and J. M. Wolfe, "Contextual cuing by global features," *J. Perception Psychophys.*, vol. 68, no. 7, pp. 1204–1216, Oct. 2006.

[28] A. Torralba, A. Oliva, M. S. Castelhano, and J. M. Henderson, "Contextual guidance of eye movements and attention in real-world scenes: The role of global features in object search," *Psychol. Rev.*, vol. 113, no. 4, pp. 766–786, Oct. 2006.

[29] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 733–740.

[30] X. Sun, H. Yao, and R. Ji, "What are we looking for: Towards statistical modeling of saccadic eye movements and visual saliency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 1552–1559.

[31] J. Zhang and S. Sclaroff, "Saliency detection: A Boolean map approach," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 153–160.

[32] Y. Li, B. Sheng, L. Ma, W. Wu, and Z. Xie, "Temporally coherent video saliency using regional dynamic contrast," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 12, pp. 2067–2076, Dec. 2013.

[33] A. Torralba, "Contextual influences on saliency," in *Neurobiology of Attention*, L. Itti, G. Rees, and J. K. Tsotsos, Eds., 1st ed. Amsterdam, The Netherlands: Elsevier, 2005, pp. 586–592.

[34] M. Cerf, J. Harel, W. Einhaeuser, and C. Koch, "Predicting human gaze using low-level saliency combined with face detection," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Vancouver, BC, Canada, 2009, pp. 241–248.

[35] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2007, pp. 1–8.

[36] C. Guo, Q. Ma, and L. Zhang, "Spatio-temporal Saliency detection using phase spectrum of quaternion Fourier transform," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.

[37] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 185–198, Jan. 2010.

[38] J. Li, M. D. Levine, X. An, X. Xu, and H. He, "Visual saliency based on scale-space analysis in the frequency domain," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 4, pp. 996–1010, Apr. 2013.

[39] A. Garcia-Diaz, V. Leborán, X. R. Fdez-Vidal, and X. M. Pardo, "On the relationship between optical variability, visual saliency, and eye fixations: A computational approach," *J. Vis.*, vol. 12, no. 6, p. 17, 2012.

[40] 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.

[41] W. Wang, Y. Wang, Q. Huang, and W. Gao, "Measuring visual saliency by site entropy rate," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 2368–2375.

[42] C. Kanan, M. H. Tong, L. Zhang, and G. W. Cottrell, "SUN: Top-down saliency using natural statistics," *Vis. Cognit.*, vol. 17, nos. 6–7, pp. 979–1003, 2009.

[43] W. Hou, X. Gao, D. Tao, and X. Li, "Visual saliency detection using information divergence," *Pattern Recognit.*, vol. 46, no. 10, pp. 2658–2669, Oct. 2013.

[44] J. Han, S. He, X. Qian, D. Wang, L. Guo, and T. Liu, "An object-oriented visual saliency detection framework based on sparse coding representations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 12, pp. 2009–2021, Dec. 2013.

[45] U. Rajashekar, L. K. Cormack, and A. C. Bovik, "Image features that draw fixations," in *Proc. Int. Conf. Image Process. (ICIP)*, vol. 2. Sep. 2003, pp. III-313–III-316.

[46] P. Reinagel and A. M. Zador, "Natural scene statistics at the centre of gaze," *Netw., Comput. Neural Syst.*, vol. 10, no. 4, pp. 341–350, 1999.

[47] D. Parikh, C. Zitnick, and T. Chen, "Determining patch saliency using low-level context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Berlin, Germany, 2008, pp. 446–459.

[48] W. Kienzle, F. A. Wichmann, B. Schölkopf, and M. O. Franz, "A non-parametric approach to bottom-up visual saliency," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2007, pp. 689–696.

[49] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, May 2001.

[50] L. Itti and C. Koch, "A saliency-based search mechanism for overt and covert shifts of visual attention," *Vis. Res.*, vol. 40, nos. 10–12, pp. 1489–1506, Jun. 2000.

[51] A. Borji, "Boosting bottom-up and top-down visual features for saliency estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 438–445.

[52] J. Li, Y. Tian, T. Huang, and W. Gao, "Cost-sensitive rank learning from positive and unlabeled data for visual saliency estimation," *IEEE Signal Process. Lett.*, vol. 17, no. 6, pp. 591–594, Jun. 2010.

[53] J. Yang and M.-H. Yang, "Top-down visual saliency via joint CRF and dictionary learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 2296–2303.

[54] E. Vig, M. Dorr, and D. Cox, "Large-scale optimization of hierarchical features for saliency prediction in natural images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 2798–2805.

[55] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2007, pp. 1–8.

[56] T. N. Vikram, M. Tscherepanow, and B. Wrede, "A saliency map based on sampling an image into random rectangular regions of interest," *Pattern Recognit.*, vol. 45, no. 9, pp. 3114–3124, Sep. 2012.

[57] A. Borji, M.-M. Cheng, H. Jiang, and J. Li, "Salient object detection: A benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5706–5722, Dec. 2015.

[58] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 3166–3173.

[59] X. Li, H. Lu, L. Zhang, X. Ruan, and M.-H. Yang, "Saliency detection via dense and sparse reconstruction," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 2976–2983.

[60] 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.

[61] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 2083–2090.

Shu Fang received the B.E. degree in software engineering from Chongqing University, Chongqing, China, in 2011. She is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Computer Science, Peking University, Beijing, China.

Her current research interests include visual saliency and brain-inspired computing.

Jia Li (M'13–SM'15) received the B.E. degree from Tsinghua University, Beijing, China, in 2005, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, in 2011.

He is currently an Associate Professor with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing. His current research interests include computer vision and image/video processing.

Yonghong Tian (M'05–SM'10) received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2005.

He is currently a Professor with the National Engineering Laboratory for Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing. He has authored or co-authored over 120 technical articles in refereed journals and conferences. His current research interests include machine learning, computer vision, and multimedia big data.

Prof. Tian is a Senior Member of Chinese Institute of Electronics, and a member of the Association for Computing Machinery and the China Computer Federation. He was a recipient of the First Prize of the 2015 Technology Innovation Award by the Chinese Institute of Electronics and the Second Prize of the 2010 National Science and Technology Progress Awards. He is an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA and the *International Journal of Multimedia Data Engineering and Management*, and a Young Associate Editor of *Frontiers of Computer Science*.

Tiejun Huang (M'01–SM'12) received the bachelor's and master's degrees in computer science from the Wuhan University of Technology, Wuhan, China, in 1992 and 1995, respectively, and the Ph.D. degree in pattern recognition and intelligent systems from the Huazhong University of Science and Technology, Wuhan, in 1998.

He is currently a Professor with the School of Electronic Engineering and Computer Science, the Chair of the Department of Computer Science, and the Director of the Institute for Digital Media Technology with Peking University, Beijing, China. His current research interests include video coding and image understanding, especially neural coding inspired information coding theory in the last years.

Prof. Huang is a member of the Board of the Chinese Institute of Electronics, the Board of Directors of the Digital Media Project, and the Advisory Board of IEEE Computing Now. He received the National Science Fund for Distinguished Young Scholars of China in 2014.

Xiaowu Chen (M'09–SM'15) received the Ph.D. degree in computer science from Beihang University, Beijing, China, in 2001.

He is currently a Professor with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University. His current research interests include virtual reality, computer graphics, and computer vision.