Region-Based Saliency Detection and Its Application in Object Recognition

Ren, Zhixiang; Gao, Shenghua; Chia, Liang-Tien; Tsang, Ivor Wai-Hung

2013

Ren, Z., Gao, S., Chia, L.-T., & Tsang, I. W.-H. (2014). Region-Based Saliency Detection and Its Application in Object Recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 24(5), 769-779.

<https://hdl.handle.net/10356/81701>

<https://doi.org/10.1109/TCSVT.2013.2280096>

© 2013 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at: [<http://dx.doi.org/10.1109/TCSVT.2013.2280096>].

Downloaded on 26 Apr 2023 21:58:38 SGT

Region-based Saliency Detection and Its Application in Object Recognition

Zhixiang Ren^{*a}, Shenghua Gao^{*b}, Liang-Tien Chia^a, and Ivor Wai-Hung Tsang^a

Abstract—The objective of this paper is twofold. First, we introduce an effective region-based solution for saliency detection. Then we apply the achieved saliency map to better encode the image features for solving object recognition task.

To find the perceptually and semantically meaningful salient regions, we extract superpixels based on an adaptive mean shift algorithm as the basic elements for saliency detection. The saliency of each superpixel is measured by using its spatial compactness which is calculated according to the results of Gaussian Mixture Model (GMM) clustering. To propagate saliency between similar clusters, we adopt a modified PageRank algorithm to refine the saliency map. Our method not only improves saliency detection through large salient region detection and noise tolerance in messy background, but also generates saliency maps with well-defined object shape. Experimental results demonstrate the effectiveness of our method.

Since the objects usually correspond to salient regions, and these regions usually play more important roles for object recognition than background, we apply our achieved saliency map for object recognition by incorporating saliency map into Sparse coding based Spatial Pyramid Matching (ScSPM) image representation. To learn a more discriminative codebook and better encode the features corresponding to the patches of the objects, we propose a weighted sparse coding for feature coding. Moreover, we also propose a saliency weighted max pooling to further emphasize the importance of those salient regions in feature pooling module. Experimental results on several datasets illustrate that our Weighted Sparse Coding based Spatial Pyramid Matching framework greatly outperforms Sparse Coding based Spatial Pyramid Matching framework, and achieves excellent performance for object recognition.

Index Terms—Saliency Detection, Superpixel, Saliency Propagation, Weighted Sparse Coding, Object Recognition.

I. INTRODUCTION

Visual saliency measures to what extent a region attracts human attention. Recently, researchers have shown an increased interest in automatic saliency detection because of its (potential) application in many scenarios, like adaptive compression [48], image retargeting [3], object detection [2], etc. However, it remains a very challenging task due to the

acquisition of prior knowledge and the difficulty in understanding and simulating the complex visual attention mechanism of humans.

In the last few decades, lots of image saliency detection algorithms have been proposed [38][37][36][57][68]. Most of these methods represent the input images in pixel-grid manner. There are two evident drawbacks for these pixel-grid based saliency detection methods. First, these methods perform poorly in the images with large salient regions. Secondly, these pixel-grid based saliency detection methods also suffer from the messy background which is very common in natural scenes. The small, sudden changes in the messy background will obtain high saliency due to the high center-surround contrast, while the homogeneous objects will get low saliency.

Recent works [17][18] suggest that “early” features, such as color, contrast and orientation, indirectly affect human attention through recognizing objects. In other words, human is attracted by objects not by individual pixels. So, it is natural to work with those perceptually meaningful image regions in saliency detection. In this paper, we propose to adopt superpixel representation for saliency detection. The concept of superpixel is proposed in [54], which is defined as the perceptually meaningful entities by grouping spatially neighboring pixels with the similar property. It is robust to noises in messy scenes and preserves the object boundary very well. Meanwhile, it is able to handle the problem of large object in other saliency methods. By leveraging superpixel technique, we propose a superpixel clustering based saliency detection technique. Our work aims at calculating saliency in the region level. In addition, motivated by the fact that the objects usually correspond to the salient regions in images [18][2][47], we apply saliency maps into object recognition task and propose a saliency inspired weighted sparse coding (WSc) to favor the salient regions for better image representation. Furthermore, to further emphasize the importance of the achieved saliency map, we also propose a saliency weighted max pooling. Both weighted sparse coding and saliency weighted max pooling contribute to the improved performance of the object recognition.

It is worth noting that our saliency calculation is different from salient object extraction [21][22][66][42], in which an existing saliency method serves to guide object extraction. In other words, these works are applications of saliency detection. In contrast, our method aims to improve saliency calculation with the help of superpixel representation.

The main contributions of this paper are summarized as follows: (1) We apply the existing techniques, i.e., superpixel

^{*} Co-first author with equal contributions to this work.

^a School of Computer Engineering, Nanyang Technological University, Singapore. ^b Advanced Digital Sciences Center, Singapore.

Corresponding Author: Shenghua Gao. Email: gaos0004@ntu.edu.sg

This study is supported by the research grant for the Human Sixth Sense Programme at the Advanced Digital Sciences Center from Singapore's Agency for Science, Technology and Research (A*STAR). We also thank Dr. Yiqun Hu and associate professor Rajan Deepu (NTU in Singapore) for their valuable suggestions.

Copyright (c) 2013 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending an email to pubs-permissions@ieee.org.

representation and PageRank algorithm, to improve saliency detection. Specifically, superpixels, the more perceptually meaningful units, are used to represent the input image, which could detect large salient objects and tolerate the outliers in the messy background during saliency detection. The PageRank algorithm is then applied to propagate saliency among similar clusters and refine our region-based saliency map. (2) We propose the saliency inspired weighted sparse coding and saliency weighted max pooling by using the achieved saliency map for solving object recognition task. Such weighted sparse coding can learn a more discriminative codebook to favor the feature coding of those more salient/important features. Moreover, to further reduce the effect of noisy background and emphasize the importance of object regions, we also propose a saliency weighted max pooling for image representation, which further improves the image representation ability.

This work is an extension of our previous work [55], which, as far as we know, is the earliest work of using superpixel for saliency detection. We mainly extend our work in the following aspects: (1) We use more datasets to fully validate the effectiveness of our saliency detection method. (2) We successfully apply our saliency map to object recognition tasks. We use the saliency detection to boost the performance of object recognition. Specifically, we first propose to use saliency inspired sparse coding to boost feature coding, then the saliency weighted max feature pooling is proposed for image representation. These methods can significantly improve the object recognition accuracy, which demonstrates the effectiveness of our saliency map.

The rest of this paper is organized as follows: In Section II, related work will be briefly discussed, including the work of saliency detection and work related to sparse coding for image classification. In Section III, we will describe our region-based saliency detection framework in details. In Section IV, experiments will be performed to evaluate our saliency detection method. In Section V, we will introduce a new sparse coding technique: Weighted sparse coding (referred to as WSc), as well as the formulations and optimization. Based on WSc, we propose to use Weighted Sparse Coding based Spatial Pyramid Matching (referred to as WScSPM) for object recognition. In Section VI, we evaluate our WScSPM on some datasets. Some important parameters are also evaluated in this section. Finally we conclude our work in Section VII.

II. RELATED WORK

A. Saliency Detection

Saliency detection methods can be divided into two categories [38]. One is top-down method, which is task-dependent and based on prior knowledge about the scenes [51][41][69], objects [61], face [40], and their interrelations [26]. Due to the prior information, better performance can be achieved by top-down methods. However, some top-down methods demand more complete understanding of the prior information, resulting in high computational costs. By contrast, bottom-up methods are easy to implement and understand. So bottom-up methods have a wider application prospect and have been

extensively adopted. Since our method belongs to the bottom-up category, we will only review the works related to bottom-up methods in details.

A common hypothesis for bottom-up saliency is that the salient stimulus is distinct from its surrounding stimuli [24]. This hypothesis is also known as “center-surround” mechanism, which consists with the performance of human vision system in the early phase. Therefore, researchers usually focus on identifying those regions with high center-surround contrast in bottom-up methods. Itti *et al.* [38] propose to determine the contrast by the Difference of Gaussians (DoG). Instead of directly calculating similarity of two pixels, Seo *et al.* [57] measure the likeness of a pixel(voxel) to its surroundings by the local regression kernels. Although these local contrast based methods perform well on most of natural images, they will fail on images with scattered background. Therefore, some methods based on global contrast are proposed. Achanta *et al.* [1] measure the saliency of each pixel by the difference between the feature of each pixel and the mean feature of the whole image in *Lab* color space. Zhai and Shah [68] measure the global contrast of a pixel by comparing it with all the other pixels in the image. Goferman *et al.* [30] model both local and global contrast by taking the positional distance into account when computing similarity between two patches.

Besides features from spatial domain, characteristics from frequency domain are also investigated for saliency detection. Hou and Zhang [37] calculate saliency on frequency domain based on spectral residual. Later, Guo *et al.* [34] point out that it is not the amplitude spectrum but the phase spectrum that contributes to saliency detection. They make use of phase spectrum to detect saliency and extend two-dimensional Fourier Transform to a quaternion Fourier Transform for spatiotemporal saliency detection.

Some approaches are based on information theory measure saliency by entropy [56], conditional entropy [45], self-information [11] and so on. The essence of these methods exploits the local complexity or uncertainty to reflect saliency. Some graph-based methods identify salient regions by equilibrium visiting frequency of each node [31][36][64].

In general, most of these approaches represent the input image in pixel-grid manner and determine the saliency of pixel according to the local/global contrast. Such representation and saliency calculation criteria may fail to detect the homogeneous and quite large salient objects. Since the pixels within the large objects have small contrast with their surrounding pixels. Moreover, the pixel representation methods is sensitive to the small sudden changes in messy background. To alleviate these problems, we advocate to use region-based solution for saliency detection and successfully apply our saliency maps to object recognition.

B. Sparse Coding for Object Recognition

Object recognition is a fundamental problem in computer vision, and it greatly relies on image representation. Many image representation models have been proposed, in which Bag-of-Words (BoW) model [59] has been fully studied and commonly adopted because of its compact representation,

Fig. 1. The framework for saliency detection. The color bars denote the GMM components. The values above the color bars are saliency values obtained by our method.

and robustness to object translation and scale variance. BoW contains the following modules: (1) Region selection and representation (feature extraction); (2) Codebook generation and feature quantization (feature coding); and (3) Histogram image representation (feature pooling).

In BoW model, usually k -means clustering is used for feature coding: the cluster centers are used as the feature codebook, and each feature is assigned to the nearest cluster center. However, such hard assignment will cause severe information loss especially for those features located around the boundary of several clusters. To avoid this, soft k -means [52] and sparse coding [65] based feature coding strategies are introduced. Compared with soft k -means, fewer parameters are used and better performance is achieved by using sparse coding, therefore sparse coding based feature coding is widely used recently. Based on sparse coding, some other sparse coding related feature coding methods [63][29][65] are also proposed and achieve good performance for image representation.

In feature pooling module, the sparse codes corresponding to all the features within each image are summarized by a vector to represent this image. In BoW model, each image is represented as a frequency histogram of the codebook entries in the whole image, it belongs to average pooling. Recent works show that max pooling demonstrates great success for sparse features [65][8]. Note that in BoW model, the pooling area is the whole image, therefore the spatial information is lost. In many cases, for object or scene images, such spatial information is very important and can greatly affect the recognition accuracy. To preserve such spatial information, Lazebnik *et al.* propose a Spatial Pyramid Matching (SPM) [43] for image representation. Specifically, in SPM, each image is partitioned into increasingly finer regions. Then Pyramid Matching Kernel [33] is used to compare corresponding sub-regions. SPM has demonstrated its effectiveness in spatial structure preserving [43][65].

C. Saliency for Object Recognition

For object recognition, the objects are usually more salient than the background. Therefore, the features located on the object should play more important roles for the recognition of the object in the images. Saliency detection has a potential application in object recognition [62][58][25]. Moosmann *et al.* [49] use the saliency of the features to boost the classifier of randomized decision trees for object recognition. Han and Vasconcelos [35] replace the first layer of the HMAX architecture with a saliency network for object recognition. Gao and Vasconcelos [25] classify the images by directly classifying

their saliency maps. Frintrop *et al.* [20] speed up classification by training the classifier based on the salient regions other than the whole image. Context information is also commonly used for object detection [16][53][50]. Shokoufandeh *et al.* [58] build a hierarchical graph structure based on saliency map to represent 3D objects. Moreover, Regions of Interests (ROI) which are defined as the class-specific regions determined by a bounding box in the image have also been applied for object recognition [7][14]. But automatically learned ROI are only available for the training data, and it cannot be used to the unseen test data. Furthermore, the ROI based method only obtains binary saliency map and cannot characterize the saliency in a consistent way, and existing studies [16][53] also show that the background also can help predict the content of the image to some extent, and they should also contribute to the image representation. In a word, all these works hint the potential application of saliency map for object recognition. But using saliency to boost the Spatial Pyramid Matching (feature extraction + feature coding + feature pooling) based object recognition is less explored.

III. SUPERPIXEL CLUSTERING FOR SALIENCY DETECTION

In this paper, we advocate superpixels as the basic units in saliency detection. We first use the adaptive mean shift algorithm to extract superpixels from the input image, then apply Gaussian Mixture Model (GMM) to cluster superpixels based on their color similarity, and finally calculate the saliency value for each cluster using spatial compactness metric together with modified PageRank propagation. Fig. 1 shows the overview of the proposed framework for salient region detection.

A. Superpixel Extraction and Clustering

Given an input image, we perform mean shift algorithm [13] in *Lab* color space to extract superpixels. Mean shift algorithm is known as nonparametric density estimator, which iteratively seeks the maxima of a density function in a set of data samples. It does not assume any prior knowledge on data cluster and has been successfully applied to many computer vision applications [15]. In mean shift algorithm, there are three parameters: R_s and R_r are the spatial and range radii of the search window respectively, R_m is the minimum point density that a resulting region must have. Similarly to [1], we use fixed parameters of 7, 20 for R_s and R_m respectively. For R_r , we set it according to the color variance of the given image:

$$R_r = \min \left(R_c, \frac{\sigma^2}{\sigma_t^2} R_c \right), \quad (1)$$

where σ^2 is the color variance of the input image (normalized by pixel number), σ_t^2 and R_c are empirical constants representing the average normalized color variance and maximum range radius respectively. In experiments, we fix σ_t^2 to 0.03¹.

It is worth noting that we adaptively set different values of R_r for different images. For complicated images (those with large color variance), R_r is set to large value so as to capture

¹It is estimated by using the average color variance of the 1,000 images available from [1]

the main color pattern but ignore insignificant variations (noise tolerance). Moreover, we set an upper bound, R_c , of range radius to avoid under-segmentation. For simple images (those with small color variance), R_r is set to small value in order to capture detailed information and avoid objects and background blending together (see Fig. 4(b)).

The mean shift process is to segment the image into homogeneous regions. However, the pixels grouped by mean shift are restricted to be connected in spatial domain, e.g. scattered background cannot be grouped together. Furthermore, the number of extracted superpixels is often large, usually in hundreds. We cannot solve this over-segmentation problems by simply modifying the parameters of mean shift without blurring the boundary between objects and background. To group similar but discontinuous regions together and alleviate the over-segmentation problem, we introduce GMM to cluster superpixels in RGB color space. After mean shift algorithm [13], every superpixel will obtain a unique RGB color [13]. Then we concatenate the RGB value of this superpixel as a 3D vector to represent the superpixel during GMM. The reason why GMM is not directly used to cluster pixels is that GMM is sensitive to the presence of outliers, especially in messy regions (such as grassland, thick tree leaves) [5]. In addition, the illumination, reflection or minor distinction of pixels may affect the robustness of GMM. Whereas mean shift algorithm is more robust to handle regions with noise [15], which is more favorable in saliency detection. Although mean shift algorithms are computationally expensive, they can greatly speed up the subsequent processing steps (especially for GMM) because of relatively few superpixels.

The number of Gaussian components is fixed in advance and K-means is used to initialize the GMM. After that, the Expectation Maximization (EM) algorithm is used to train GMM parameters [5]. Finally, the probability of the i^{th} superpixel SP_i belonging to the k^{th} cluster is calculated as

$$P_k(SP_i) = \frac{\pi_k \cdot \mathcal{N}(SP_i | \mu_k, \Sigma_k)}{\sum_{k=1}^K \pi_k \cdot \mathcal{N}(SP_i | \mu_k, \Sigma_k)} \quad (2)$$

where π_k is the mixing coefficient of the k^{th} Gaussian components whose mean vector is μ_k and covariance matrix is Σ_k .

B. Salient Region Detection

Intuitively, background regions have larger spread in spatial domain compared to salient regions, i.e., the more compact the clusters are spread, the more salient they will be. Gopalakrishnan *et al.* [32] proposed compactness metric to evaluate the spread of cluster in the spatial domain. In their method, the cluster compactness is measured based on inter-cluster distances defined as

$$COMP_k = \left(\sum_{j=1}^K d_{kj} \right)^{-1}, \quad (3)$$

where d_{kj} is the distance between the k^{th} cluster and the j^{th} cluster. The inter-cluster distance d_{kj} is defined as

$$d_{kj} = \frac{\sum_y \|Y - C_j\|^2 \cdot P_k(x)}{\sum_x P_k(x)}, \quad (4)$$

where Y denotes the spatial coordinates of pixel y , and C_j denotes the spatial center of the j^{th} cluster. We adopt the same compactness metric except the probability $P_k(y)$. In our model, if pixel $y \in SP_i$, we let $P_k(y) = P_k(SP_i)$ in Equation (4). Finally, the saliency of the k^{th} cluster is defined as

$$\tilde{S}_k = COMP_k. \quad (5)$$

C. Saliency Refinement with Propagation

For clustering algorithm, it is difficult to choose the correct number of clusters K . We conservatively fix $K = 8$ in our experiments. However, in some situation, the perceptually meaningful regions are less than the cluster number, thus some regions which should belong to one cluster will be grouped into several clusters. Take Fig. 1 for example, the grassland has two similar clusters in green, and the tiger is also grouped into several similar clusters. While the compactness metric is based on the spread of a cluster. If one (perceptually meaningful) cluster is over-segmented into several sub-clusters, the compactness may be highly distorted. Therefore, we propose a modified PageRank algorithm to propagate saliency between similar clusters.

In web page link analysis, PageRank algorithm is applied to measure the relative importance of a page [10]. The original PageRank Equation is given as follows:

$$PR(p_i) = (1 - \lambda) \frac{1}{N} + \lambda \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)} \quad (6)$$

where $PR(p_i)$ is called the PageRank of any given page p_i . $M(p_i)$ is the set of pages that link to p_i , $L(p_j)$ is the number of outbound links on page p_j , λ is damping factor and N is the total number of pages. PageRank is widely used in label propagation for semi-supervised learning [60][23]. Saliency propagation is also a semi-supervised learning problem. So it is nature to use PageRank algorithm to refine saliency. From Equation (6), a page linked by many pages with high PageRank receives a high rank itself. In saliency refinement, we treat a cluster as a web page, and assume that a cluster, which is similar to many other clusters with high saliency, should also receive a high saliency. Under this assumption, we refine cluster saliency by propagating them between clusters based on following pairwise similarity:

$$A_{ij} = \begin{cases} \exp \left(- \frac{\|\mu_i - \mu_j\|^2}{2\sigma_\mu^2} \right) & \text{if } i \neq j, \\ 0 & \text{otherwise,} \end{cases} \quad (7)$$

where $\mu_i \in \mathbb{R}^{3 \times 1}$ is the mean color vector of the i^{th} cluster, σ_μ^2 is the sum of the variance of each dimension of μ . Similar to the PageRank algorithm, we define our saliency propagation formula as follows:

$$S_k = (1 - \lambda) \tilde{S}_k + \lambda \sum_{i=1}^K \frac{S_i * A_{ik}}{\sum_{j=1}^K A_{jk}}, \quad (8)$$

where $\tilde{S}_k$ is the initial saliency value computed from the compactness of the k^{th} cluster (see Section III-B), and λ is a trade-off parameter between the initial saliency and

the saliency contributed from other clusters (set to 0.5 in experiments).

It should be noted that it is the first time to apply PageRank in saliency detection and we modify it in three aspects. (1) Every page is uniformly assigned a prior probability $\frac{1}{N}$ in the PageRank formula, while we assign the initial saliency $\tilde{S}$ as the prior saliency for each cluster. (2) The PageRank algorithm uses binary transition matrix, while we use similarity weighted transition matrix. (3) The transition contribution of one page is normalized by its outbound links in original PageRank, while we normalize the saliency of a cluster by its inbound links. This is because using the outbound normalization will bias toward the clusters which have more similar clusters.

Fig. 2. Refinement propagates saliency between similar colors and leads to enhanced saliency maps (images are from the EPFL dataset).

We solve Equation (8) directly by analytical method. In matrix form, the refined saliency vector S is

$$S = (1 - \lambda)(I - \lambda A)^{-1} \tilde{S}, \quad (9)$$

where I is the identity matrix, $\tilde{S}$ denotes the initial cluster saliency vector. After the saliency values of clusters are calculated, the saliency of superpixels can be obtained by accumulating contributions from all clusters according to its cluster distribution $P_k(SP_i)$:

$$Sal(SP_i) = \exp \left(\sum_{k=1}^K S_k P_k(SP_i) \right), \quad (10)$$

where the exponential function is used for contrast stretching. Note that it cannot improve saliency detection, but just gives a better visual effect.

Fig. 2 compares the saliency maps with refinement and without refinement. We can see that, the refinement strategy can propagate saliency between similar colors. Before propagation, the wool ball, red bottle cap and the dog are partially salient. In contrast, they become almost entirely salient after propagation. The last row shows an example of background suppression. Due to most of the poles get low saliency values, this trend is propagated to the small parts of poles with high saliency and thus suppresses the background.

IV. EXPERIMENTS FOR SALIENCY DETECTION

We conduct experiments on three commonly used, region-based datasets. The **EPFL** dataset [1]²: it contains 1,000

images with accurate object-contour based ground truth. The **CMU** dataset [4]³: it is comprised of 38 groups (643 images) with hand-annotated ground-truth. The **MSRA** dataset [46]⁴: it includes 5,000 images as well as the corresponding rectangular annotations of salient region. Some eye tracking datasets [40][11] are also widely used in saliency evaluation. In such datasets, only several fixation points are recorded, which are not sufficient to indicate the complete salient regions. To alleviate this problem, most of methods [40][11] usually convolve the eye tracking data with a 2D Gaussian filter to get rough salient regions as ground truth. Due to this fact, we only evaluate different methods on the region-based datasets.

We compare our method with previous methods including Itti's method (ITTI) [38], Spectral Residual method (SR) [37], Graph-Based saliency method (GB) [36], Frequency-Tuned method (FT) [1] the salient region detection method based on color and orientation distributions (COD) [32], Region Contrast method (RC) [12] and Context-Based method (CB) [39]. Note that the last two methods are more related to our method which also calculate saliency with the help of some segmentation algorithms. Since most of saliency evaluation datasets are region-based, Similar to [64][36][40], the average Receiver Operating Characteristic (ROC) curve is used to estimate the performances of different methods. By sweeping a threshold from 0 to 1 over the normalized saliency maps, we calculate the true positive rate and false positive rate and draw the ROC curve. We also use another commonly used metric: linear Correlation Coefficient (CC) [6] to evaluate the performance. CC measures the strength of a linear relationship between the saliency map and the groundtruth map. Fig. 3 shows the performance of all methods on these three datasets in terms of the ROC curves and CC. Performance rankings of different methods are not the same over ROC and CC scores, but it is worthy of noting that region-based methods generally perform better than pixel-grid based methods on all the datasets, except GB on the MSRA dataset. This indicates the superpixel representation proposed by us firstly has great value in region-based saliency detection. It is also validated by some neurobiological finding that objects predict fixations better than early saliency [17][18]. We also calculate the Area Under the ROC Curve (AUC) of different methods in Table I. Our method with superpixel representation is superior than the five pixel-grid based methods (ITTI, SR, GB, FT and COD) and is comparable to the other two region-based methods (RC and CB) on all the datasets. Moreover, our method achieves the best performance in terms of AUC in the MSRA dataset.

To obtain a visual comparison among the different approaches, we show some examples in Fig. 4. Since the results generated by region-based methods are very similar, we only give our saliency maps here. As discussed in section I, most of the compared methods fail to detect the large salient objects such as the tiger, ape and leaf in the figure 3(a), 3(b) and 3(e). In contrast, our method generates more meaningful saliency maps with well-defined object boundaries. It is worth noting

³<http://chenlab.ece.cornell.edu/projects/touch-coseg/>

⁴http://research.microsoft.com/en-us/um/people/jiansun/salientobject/salient_object.htm

²http://ivrg.epfl.ch/supplementary_material/RK_CVPR09/index.html

TABLE I
AREAS UNDER AVERAGE ROC CURVES (AUC) OF ALL METHODS.

Method	SR	ITTI	GB	FT	COD	RC	CB	Ours
EPFL	0.768	0.787	0.872	0.884	0.875	0.966	0.967	0.964
CMU	0.788	0.745	0.833	0.804	0.814	0.913	0.875	0.883
MSRA	0.740	0.764	0.854	0.659	0.821	0.846	0.851	0.893

Fig. 3. Performance of the compared methods over three datasets in terms of average ROC curves and Correlation Coefficient (CC).

that despite applying the same GMM model and compactness metric, the COD method generates saliency maps with rather poor quality compared with ours, which in some extent proves the effectiveness of the superpixel representation and propagation strategy. Moreover, our method outperforms other methods especially for complicated images (such as Fig. 4(a)) or images with texture background (such as Fig. 4(d) and (e)).

V. WEIGHTED SPARSE CODING BASED SPATIAL PYRAMID MATCHING FOR OBJECT RECOGNITION

As shown in Fig. 4, objects in the image usually correspond to salient regions. By leveraging the calculated saliency map, we propose a saliency inspired weighted sparse coding technique for feature coding, and a saliency weighted max pooling for feature pooling. Specifically, our weighted sparse coding based object recognition framework can be described as follows: (1) After extracting the features, we use weighted

Fig. 4. The comparison of saliency maps generated from different methods. The first row from left to right: the input images, SR method [37], GB method [36], COD method [32]. The second row: the ground truth, ITTI's method [38], FT method [1], our method. All the images come from the EPFL dataset.

sparse coding for feature coding. (2) We adopt saliency weighted max pooling for image representation. To preserve the spatial information, SPM [43] technique is also adopted. (3) We use SVM to learn classifiers on the training images and predict the class label of the test image. We show the role of saliency map in image representation in Fig. 5. As the SVM is commonly used to train the classifier and predict the label of the test image in all the image classifications, similar to previous work [43][65][28], we just focus on explaining how to get an effective image representation based on the weighted sparse coding and saliency weighted max pooling, and will not explain the SVM in details.

A. Saliency Inspired Weighted Sparse Coding for Local Feature Coding

Denote X as a collection of SIFT features: $X = [x_1, \dots, x_N]$, in which each column x_i ($x_i \in \mathbb{R}^{d \times 1}$) corresponds to one feature extracted from a 16×16 image patch, and N is the number of SIFT features, i.e., there are N patches in all. Denote the codebook as $U = [u_1, \dots, u_k] \in \mathbb{R}^{d \times k}$. Therefore k -means based feature coding can be formulated as follows:

$$\min_{U, V} \sum_{i=1}^N \|x_i - Uv_i\|^2 \quad \text{s.t.} \quad \text{Card}(v_i) = 1, \|v_i\|_1 = 1, \forall i. \quad (11)$$

Here $V = [v_1, \dots, v_N]$ is the cluster index matrix. Each v_i ($v_i \in \mathbb{R}^{k \times 1}$) has only one non-zero entry 1. Such hard

Fig. 5. The flowchart in the top blue rectangle is sparse coding based image representation. The flowchart in the bottom red rectangle is saliency map boosted image representation. Saliency map boosts the feature coding (weighted sparse coding) and feature pooling (saliency weighted max pooling).

assignment may cause information loss, and it is especially not suitable for those features located at the boundary of several clusters. To reduce the information loss during the feature coding, the constraint that each feature is only assigned to one cluster ($Card(v_i) = 1$) is relaxed for the soft assignment method [52]. Furthermore, following the soft assignment, we also allow each feature to belong to several clusters. Therefore, we add the sparsity constraint on the weight vector v_i to the objective of feature coding. We arrive at the objective of sparse coding:

$$\min_{U,V} \sum_{i=1}^N (\|x_i - Uv_i\|^2 + \gamma \|v_i\|_1) \quad \text{s.t.} \quad \|u_j\| = 1, \forall j. \quad (12)$$

In sparse coding based feature coding process, all the features have the same weight, therefore they have the equal contributions to the codebook learning and feature coding. However as we know, the image patches located in the object contribute more to the recognition of the object than the surrounding patches [2][47]. As a result, we desire to learn a more discriminative codebook, which prefers to reduce the feature quantization loss for those more important features. In this way, more useful information will be kept in feature coding process, and the image representation can be enhanced. To this end, we propose two weighted sparse coding formulations for feature coding and codebook learning.

We denote the feature to be encoded as x_i , and denote the weight of the patch this feature corresponding to as w_i . The weight of this patch reflects the importance of this feature, therefore we simply set w_i be proportional to the saliency of the patch. To reduce the feature quantization error for those more important features, we use w_i to weight the feature quantization error term corresponding to feature x_i in Equation (12), and arrive at the objectives of our WSc in Equation (13) and Equation (14).

$$\min_{U,V} \sum_i w_i (\|x_i - Uv_i\|_F^2 + \gamma \|v_i\|_1) \quad \text{s.t.} \quad \|u_m\| = 1 \quad (13)$$

$$\min_{U,V} \sum_i (w_i \|x_i - Uv_i\|_F^2 + \gamma \|v_i\|_1) \quad \text{s.t.} \quad \|u_m\| = 1 \quad (14)$$

In Equation (13), we impose the weight on both the reconstruction error and the sparsity term because these two terms jointly correspond to the feature coding of a given feature. We desire those important features to have larger contributions in

feature coding and codebook learning process. Therefore we add weight on the feature coding process. In Equation (14), we only impose the weight on the reconstruction error because the important features may have smaller reconstruction error. In the following sections, we denote the objectives of WSc in Equation (13) and Equation (14) as WSc-I and WSc-II respectively.

Both WSc-I and WSc-II are not convex for both U and V , but they are convex for the codebook U (the sparse codes V) when the sparse codes V (the codebook U) is fixed. Following the previous works [65][44], we optimize the codebook and the sparse codes alternatively. Specifically, when the sparse codes of all the features are fixed, we use the Lagrange Dual method to optimize the codebook [44][65]. When the codebook U is learned, we use feature-sign search algorithm [44] to optimize the sparse codes. In practice, the number of the features is large-scale, and it is impossible to perform feature coding for all features simultaneously. Following previous work [65], we sample some features from the whole feature pool to learn the codebook, then use the learned codebook to encode all the features.

B. Feature Pooling and Image Representation

After encoding each feature to a sparse code vector, we also use the max pooling for image representation [8][9][65]. Moreover, to further reduce the effect of background and emphasize those salient object regions in image representation, we also incorporate such saliency information into the max pooling process, and propose a saliency weighted max pooling. Specifically, if a certain region/patch is more salient, it is more probably to be the object. As a result, this region/patch should contribute more to the representation of the image, and it is natural that we assign larger weights to the sparse codes corresponding to the more salient patches, and the weight used for weighting the sparse codes should be proportional to the weight of this patch. For simplification, we just fix the weight to be the same with the weight of this region.⁵ Suppose an image contain M patches, and the codebook size is k , then after saliency weighted max pooling, this image region will be represented by a k dimensional feature y . The j^{th} entry of y is calculated by the following equation:

$$y_j = \max\{w_1|v_{1j}|, w_2|v_{2j}|, \dots, w_M|v_{Mj}|\} \quad (15)$$

⁵We can also use other function of the weight of the patch to weight the sparse codes of this patch. The study of using more complex function of this weight to weight the sparse codes is beyond the study scope of this paper.

The spatial structure is a very important factor for object recognition. To preserve such spatial information, SPM model is also incorporated into our WScSPM framework. Following the setting of ScSPM [65], we use 3-layer SPM in our implementation. We divide the image into 1×1 , 2×2 , and 4×4 subregions respectively in each layer, and saliency weighted max pooling is performed within each subregion. We also set the same weights for different layers in SPM.

C. Weight Determination in Weighted Sparse Coding

The way of determining the weight for feature is vital in WSc, and it affects the quality of the learned codebook and feature pooling. As aforementioned, the patches which cover the objects are more important, therefore their weights should be larger.

For a patch P_i which is represented by feature x_i , we take average saliency value of all the pixels within this patch as the saliency value of this patch, and denote it as Sal_i ($Sal_i \in [0, 1]$). A linear mapping function is used to map the saliency value of each patch to its weight in WSc:

$$w_i = (1 - w_0) \times Sal_i + w_0 \quad (16)$$

Here w_0 ($0 < w_0 < 1$) is the minimum weight of all the features in WSc. In practice, in some cases, background may be useful to the object recognition [16][53]. What's more, sometimes the salient map is not very good. As a result some parts of the objects are not salient, and their saliency values may be 0. Nevertheless, they also should contribute to the codebook learning. Regarding these reasons, we set the minimum weights of all the features to some non-zero value w_0 . When $w_0 = 1$, it comes to the case that all the features have the same weights. In such case our WScSPM degenerates to ScSPM.

VI. EXPERIMENTS FOR OBJECT RECOGNITION

In our experiments, we use four datasets to evaluate our method: (1) The 17 Flower Category Database (Flower-17)⁶; (2) The 102 Flower Category Database (Flower-102)⁷; (3) The ImageNet-Animal&Insect (ImageNet-AI) dataset⁸; (4) The Pascal VOC 2006 dataset⁹.

A. Experimental Setting

Following previous works [65][43][27], we adopt commonly used SIFT feature for local patch characterization. Specifically, the SIFT features are extracted in dense grid regions, whose patch size and the step between the centers of two neighboring patches is 16 and 4 respectively. For efficiency, we reduce the image size by $\frac{1}{2}$ on both height and width for the Flower-17 and Flower-102 dataset, and we

reduce the max side (width or height) of each image to 300 pixels for the ImageNet-AI and Pascal VOC 2006 dataset with the original aspect ratio. For the codebook size, following the work of Yang *et al.* [65], the size of the codebook is fixed to 1024, and around $(0.6 - 1.0) \times 10^5$ features are randomly sampled to generate the codebook for each dataset. Following the work of ScSPM [65], the weight of sparsity term (γ) is fixed to be 0.3 for all the datasets. In our experiments, w_0 is set as 0.5 for simplification on all the datasets. We also randomly generate 5 training/validation/test splits for the ImageNet-AI dataset, and we fix the training/validation image number as 40/35, and use all the rest images as test data. All the results of different methods reported in the paper are based on these standard splits and the same features. To keep consistent with the work of Yang *et al.* [65], we also adopt linear SVM to learn the classifier because of its fast speed and good performance for the classification of sparse coding based image representation. We use the average classification accuracy for evaluation [65].

B. Performance Comparison

The performance comparisons between different methods on the Flower-17, Flower-102, and ImageNet-AI datasets are reported in Table II. We can see that on these datasets, our method outperforms the baseline method, ScSPM, by about 6%, 7%, and 5% in terms of recognition accuracy, respectively. Moreover, we also conduct t-test to compare our method with ScSPM. Under the commonly used criteria (alpha level is set to be 5%), this improvement of our method over ScSPM is considered to be very statistically significant. Further, we list the performance of our method on the Pascal VOC dataset in Table III. Since each image may contain more than one label in this dataset, following the commonly used criteria, we also use the Average Precision (AP) for evaluation [19].¹⁰ We can see our method outperforms the ScSPM, and also achieves the best performance in terms of average AP. Needless to say, our saliency weighted feature coding and saliency weighted max pooling can also be plugged into other feature encoding methods, like Kernel Sparse Representation, Locality-constrained Learning Coding, *etc.*, seamlessly. For simplification, here we do not show the performance of these methods based such saliency weighted feature coding and pooling.

From Table II we can also see that our WScSPM based methods outperform the baseline methods on all datasets and achieve the best performance¹¹. Specifically, from the comparisons between WScSPM-I+mp, WScSPM-II+mp and ScSPM+mp, we can see that WScSPM achieves better performance than ScSPM, which proves the effectiveness of

¹⁰Here the AP is calculated based on PASCAL VOC 2010 Evaluation Protocol [19], and it is calculated as follows, “(1): Compute a version of the measured precision/recall curve with precision monotonically decreasing, by setting the precision for recall r to the maximum precision obtained for any recall $r' > r$. (2): Compute the AP as the area under this curve by numerical integration. No approximation is involved since the curve is piecewise constant.”

¹¹For simplification, we denote our WScSPM based on WSc-I and WSc-II as WScSPM-I and WScSPM-II respectively. We also denote the WScSPM with max pooling as WScSPM+mp, and we denote the WScSPM with saliency weighted max pooling as WScSPM+smp.

⁶www.robots.ox.ac.uk/~vgg/data/flowers/17/index.html

⁷www.robots.ox.ac.uk/~vgg/data/flowers/102/index.html

⁸We download a subset of ImageNet (www.image-net.org/), and it contains 33 categories which are all animals and insects. We download 200 images for each category. After removing those gray scale images and images with too large/small aspect ratio ($\maxSide/\minSide > 1.5$), the images for each category ranges from 139 to 197, and the total images in ImageNet-AI is 5853.

⁹<http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2006/>

TABLE III
AP OF DIFFERENT METHODS ON THE PASCAL VOC 2006 DATASET (%).

	bicycle	bus	car	cat	cow	dog	horse	motorbike	person	sheep	Average
ScSPM+mp [65]	77.47	79.63	87.28	59.19	58.43	39.18	52.77	64.72	53.86	70.83	64.34
WScSPM-I+mp	78.34	81.19	87.25	62.82	60.53	40.02	50.33	63.61	55.30	71.21	65.06
WScSPM-II+mp	79.11	80.27	87.33	60.81	58.42	40.98	51.97	67.94	54.78	71.56	65.32

TABLE II
AVERAGE CLASSIFICATION ACCURACY OF DIFFERENT METHODS (%).

Dataset	Flower-17	Flower-102	ImageNet-AI
SPM [43]	67.64±3.57	45.55	37.80±0.51
MTJSRC [67]	68.43±1.03	42.91	36.37±0.19
ScSPM+mp [65]	71.18±2.56	47.04	39.80±0.83
ScSPM+mp	75.39±2.96	50.22	43.82±0.79
WScSPM-I+mp	72.06±4.86	47.46	41.47±0.94
WScSPM-I+mp	76.76±3.67	54.23	43.85±1.04
WScSPM-II+mp	72.35±3.06	49.64	42.76±1.35
WScSPM-II+mp	76.76±3.24	53.29	45.02±1.02

Fig. 6. The first row shows some images from the Flower-17 dataset. These images are misclassified by ScSPM+mp, but can be correctly classified by WScSPM-I+mp. The second row gives their corresponding saliency maps.

WSc in codebook learning, and such discriminative codebook does improve the image classification performance. By using saliency weighted max pooling, the performance of WScSPM can be further improved. We also try to combine ScSPM with saliency weighted max pooling, and the performance of ScSPM can be further improved, but is still lower than our WScSPM+mp, which proves the effectiveness of weighted sparse coding. Moreover, WScSPM-II usually achieves better performance than WScSPM-I when only max pooling is used. After the saliency weighted max pooling, their performances are similar. We also show some images in Fig. 6. They are misclassified by ScSPM, while can be correctly classified by our WScSPM-I+mp. We can see that although the background of these images is messy, the objects are highlighted in the saliency maps, which will obviously boost the performance of object recognition.

Compared with the Flower datasets, the ImageNet-AI dataset and the Pascal VOC 2006 dataset are more challenging. On these two datasets, there are severe variances in occlusion, illumination, view angle, *etc.* Moreover, on VOC 2006, there are more than one objects in each image sometimes, which is a challenging for saliency detection method. But our method still demonstrates the best performance, which proves the effectiveness of our method. We conjecture that our method can achieve even better performance if we integrate our saliency method with other co-saliency detection methods, which detects objects frequently appearing in a collection of data.

C. Parameter Evaluation

w_0 is a critical parameter for weight learning. It determines the way of mapping saliency values to feature weights. In this subsection, we investigate the effect of different w_0 for object recognition on the Flower-17 dataset.

We plot the relationship between w_0 and object recognition accuracy for WScSPM-I/WScSPM-II + mp/smp in Fig. 7. It shows that the performance of WScSPM is stable when w_0 in the range of 0.3 – 0.6. We also note that the performance of WScSPM based methods usually outperform ScSPM (71.18%) for most w_0 . This figure also shows that the saliency weighted max pooling always outperforms max pooling for object recognition, and this proves the effectiveness of our saliency weighted max pooling strategy.

Fig. 7. The effect of w_0 on object recognition on the Flower-17 dataset. We can see that saliency weighted max pooling can further improve the object recognition accuracy.

VII. CONCLUSION

This paper proposes a promising saliency detection approach which can generate accurate saliency maps with well-defined object boundary. To avoid deficiency of pixel representation in large salient region detection and tolerate noise, our method employs adaptive mean shift algorithm to extract perceptually and semantically meaningful superpixels as features. We also refine the saliency values by a modified PageRank algorithm for propagating saliency between similar clusters. Experiments show that these two strategies can improve the saliency detection. Furthermore, we also apply our saliency map for object recognition. To better encode those salient regions which usually correspond to the objects, we propose a feature importance based feature coding technique: weighted sparse coding. The weighted sparse coding can learn a discriminative codebook which favors the coding of those more important object regions. We also propose a saliency weighted max pooling for image representation. Experimentally results show that our weighted sparse coding can outperform sparse coding for object recognition, and this proves the effectiveness of our weighted sparse coding.

REFERENCES

- [1] R. Achanta, S. H. F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2009.
- [2] B. Alexe, T. Deselaers, and V. Ferrari. What is an object. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [3] S. Avidan and A. Shamir. Seam carving for content-aware image resizing. *ACM Transactions on Graphics, SIGGRAPH 2007*, 26(3), July 2007.
- [4] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen. icoseg: Interactive co-segmentation with intelligent scribble guidance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2010.
- [5] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 1 edition, 2007.
- [6] A. Borji, D. N. Sihite, and L. Itti. Quantitative analysis of human-model agreement in visual saliency modeling: A comparative study. *IEEE Transactions on Image Processing*, 22(1):55–69, 2013.
- [7] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. In *Proceeding of the International Conference on Image and Video Retrieval*, 2007.
- [8] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [9] Y.-L. Boureau, J. Ponce, and Y. LeCun. A theoretical analysis of feature pooling in visual recognition. In *The International Conference on Machine Learning*, 2010.
- [10] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *The World Wide Web*, April 1998.
- [11] N. Bruce and J. Tsotsos. Saliency based on information maximization. In *Advances in Neural Information Processing Systems 19*, pages 155–162, Cambridge, MA, December 2006.
- [12] M. Cheng, G. Zhang, N. J. Mitra, X. Huang, and S. Hu. Global contrast based salient region detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 409–416, June 2011.
- [13] C. M. Christoudias, B. Georgescu, and P. Meer. Synergism in low level vision. In *Proceedings of the International Conference on Pattern Recognition*, August 2002.
- [14] O. Chum and A. Zisserman. An exemplar model for learning object classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
- [15] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:603–619, 2002.
- [16] S. K. Divvala, D. Hoiem, J. H. Hays, A. A. Efros, and M. Hebert. An empirical study of context in object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.
- [17] W. Einhäuser, M. Spain, and P. Perona. Objects predict fixations better than early saliency. *Journal of Vision*, 8(14):1–26, 2008.
- [18] L. Elazary and L. Itti. Interesting objects are visually salient. *Journal of Vision*, 8(3):1–15, 2008.
- [19] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results. <http://www.pascal-network.org/challenges/VOC/voc2010/workshop/index.html>.
- [20] S. Frintrop, A. Nüchter, H. Surmann, and J. Hertzberg. Saliency-based object recognition in 3d data. In *the International Conference on Intelligent Robots and Systems*, 2004.
- [21] Y. Fu, J. Cheng, Z. Li, and H. Lu. Saliency cuts: an automatic approach to object segmentation. In *Proceedings of the International Conference on Pattern Recognition*, 2008.
- [22] K. Fukuchi, K. Miyazato, A. Kimura, S. Takagi, and J. Yamato. Saliency-based video segmentation with graph cuts and sequentially updated priors. In *Proceedings of the IEEE international conference on Multimedia and Expo*, pages 638–641, 2009.
- [23] B. Gao, T.-Y. Liu, W. Wei, T. Wang, and H. Li. Semi-supervised ranking on very large graphs with rich metadata. In *Proceedings of the 17th International Conference on Knowledge Discovery and Data Mining*, pages 96–104, 2011.
- [24] D. Gao, V. Mahadevan, and N. Vasconcelos. The discriminant center-surround hypothesis for bottom-up saliency. In *Advances in Neural Information Processing Systems 20*, pages 497–504, December 2007.
- [25] D. Gao and N. Vasconcelos. Discriminant saliency for visual recognition from cluttered scenes. In *Advances in Neural Information Processing Systems 17*, pages 481–488, December 2004.
- [26] D. Gao and N. Vasconcelos. Integrated learning of saliency, complex features, and object detectors from cluttered scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 282–287, June 2005.
- [27] S. Gao, L.-T. Chia, and I. W.-H. Tsang. Multi-layer group sparse coding for concurrent image classification and annotation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
- [28] S. Gao, I. W. Tsang, and L.-T. Chia. Sparse representation with kernels. *IEEE Transactions on Image Processing*, 22(2):423–434, 2013.
- [29] S. Gao, I. W.-H. Tsang, and L.-T. Chia. Laplacian sparse coding, hypergraph laplacian sparse coding, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99(PrePrints), 2012.
- [30] S. Goferman, L. Zelnik-Manor, and A. Tal. Context-aware saliency detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2010.
- [31] V. Gopalakrishnan, Y. Hu, and D. Rajan. Random walks on graphs to model saliency in images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1698–1705, June 2009.
- [32] V. Gopalakrishnan, Y. Hu, and D. Rajan. Salient region detection by modeling distributions of color and orientation. *IEEE Transaction on Multimedia*, 11(5):892–905, August 2009.
- [33] K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In *Proceedings of the International Conference on Computer Vision*, 2005.
- [34] C. Guo, Q. Ma, and L. Zhang. Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2008.
- [35] S. Han and N. Vasconcelos. Biologically plausible saliency mechanisms improve feedforward object recognition. *Vision Research*, 50(22):2295–2307, 2010.
- [36] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. In *Advances in Neural Information Processing Systems 20*, pages 545–552, Cambridge, MA, December 2007.
- [37] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2007.
- [38] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, November 1998.
- [39] H. Jiang, J. Wang, Z. Yuan, T. Liu, N. Zheng, and S. Li. Automatic salient object segmentation based on context and shape prior. In *British Machine Vision Conference*, 2011.
- [40] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In *Proceedings of the International Conference on Computer Vision*, 2009.
- [41] C. Kanan, M. H. Tong, L. Zhang, and G. W. Cottrell. Sun: Top-down saliency using natural statistics. *Visual Cognition*, 17:979–1003, August 2009.
- [42] C.-C. Kao, J.-H. Lai, and S.-Y. Chien. Automatic object segmentation with salient color model. In *Proceedings of the IEEE international conference on Multimedia and Expo*, 2011.
- [43] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, USA, 17-22 June 2006*, pages 2169–2178.
- [44] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *Advances in Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 4-7, 2006*, pages 801–808.
- [45] Y. Li, Y. Zhou, J. Yan, Z. Liu, and J. Yang. Visual saliency based on conditional entropy. In *Proceedings of the Asian Conference on Computer vision*, 2009.
- [46] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2007.
- [47] Y. Luo, J. Yuan, P. Xue, and Q. Tian. Saliency density maximization for efficient visual objects discovery. *IEEE Transactions on Circuits and Systems for Video Technology*, 21:1822–1834, 2011.
- [48] L. Marchesotti, C. Cifarelli, and G. Csurka. A framework for visual saliency detection with applications to image thumbnailing. In *Proceedings of the International Conference on Computer Vision*, 2009.
- [49] F. Moosmann, D. Larlus, and F. Jurie. Learning saliency maps for object categorization. In *The European Conference on Computer Vision Workshop on the Representation and Use of Prior Knowledge in Vision*, 2006.

- [50] A. Oliva and A. Torralba. The role of context in object recognition. *Trends in Cognitive Sciences*, 11:520–527, December 2007.
- [51] A. Oliva, A. Torralba, M. S. Castelano, and J. M. Henderson. Top-down control of visual attention in object detection. In *Proceedings of the IEEE Conference on Image Processing*, pages 253–256, September 2003.
- [52] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, Minnesota, USA, 18-23 June 2007*.
- [53] O. Ramström and H. I. Christensen. Object detection using background context. In *Proceedings of the International Conference on Pattern Recognition*, 2004.
- [54] X. Ren and J. Malik. Learning a classification model for segmentation. In *Proceedings of the International Conference on Computer Vision*, October 2003.
- [55] Z. Ren, Y. Hu, L.-T. Chia, and D. Rajan. Improved saliency detection based on superpixel clustering and saliency propagation. In *Proceedings of the ACM international conference on Multimedia*, 2010.
- [56] L. W. Renninger, J. M. Coughlan, P. Verghese, and J. Malik. An information maximization model of eye movements. In *Advances in Neural Information Processing Systems 17*, pages 1121–1128, Cambridge, MA, 2004.
- [57] H. J. Seo and P. Milanfar. Static and space-time visual saliency detection by self-resemblance. *Journal of Vision*, 9(12):1–17, 2009.
- [58] A. Shokoufandeh, I. Marsic, and S. J. Dickinson. View-based object recognition using saliency maps. *Image and Vision Computing*, 17:445–460, April 1999.
- [59] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, pages 1470–1477, 2003.
- [60] J. Tang, X.-S. Hua, G.-J. Qi, and X. Wu. Typicality ranking via semi-supervised multiple-instance learning. In *Proceedings of the ACM international conference on Multimedia*, pages 297–300, 2007.
- [61] A. Torralba, A. Oliva, M. S. Castelano, and J. M. Henderson. Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search. *Psychological Review*, 113(4):766–786, October 2006.
- [62] D. Walther, U. Rutishauser, C. Koch, and P. Perona. On the usefulness of attention for object recognition. In *The European Conference on Computer Vision*, 2004.
- [63] J. Wang, J. Yang, K. Yu, F. Lv, and Y. Gong. Learning locality-constraint linear coding for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [64] W. Wang, Y. Wang, Q. Huang, and W. Gao. Measuring visual saliency by site entropy rate. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2368–2375, June 2010.
- [65] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Florida, USA, June 2009*.
- [66] H. Yu, J. Li, Y. Tian, and T. Huang. Automatic interesting object extraction from images using complementary saliency maps. In *Proceedings of the ACM international conference on Multimedia*, pages 891–894, Firenze, Italy, 2010.
- [67] X.-T. Yuan and S. Yan. Visual classification with multi-task joint sparse representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [68] Y. Zhai and M. Shah. Visual attention detection in video sequences using spatiotemporal cues. In *Proceedings of the ACM international conference on Multimedia*, pages 815–824, Santa Barbara, CA, October 2006.
- [69] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell. Sun: A bayesian framework for saliency using natural statistics. *Journal of Vision*, 8(7):1–20, 2008.

Zhixiang Ren received the B.E. degree in computer science and technology from the Xidian University in 2006, and M.S.E. degree from Chinese Academy of Sciences in 2009. She is currently pursuing the Ph.D. degree at the School of Computer Engineering, Nanyang Technological University. Her research interests include multimedia signal processing, content/perceptual-based video analytics, computer vision, machine learning etc.

Shenghua Gao received the B.E. degree from the University of Science and Technology of China in 2008, and received the Ph.D. degree from the Nanyang Technological University in 2013. He is currently a postdoc in Advanced Digital Sciences Center, Singapore. He was awarded the Microsoft Research Fellowship in 2010. His research interests include computer vision and machine learning.

Liang-Tien Chia received the B.S. and Ph.D. degrees from Loughborough University (of Technology), Loughborough, U.K., in 1990 and 1994, respectively. He is an Associate Professor in the School of Computer Engineering, Nanyang Technological University, Singapore. He was the Director, Centre for Multimedia and Network Communications from 2002-2007 and Head, Division of Computer Communications during 2007-2010. His research interests can be broadly categorized into the following areas: Internet-related research with emphasis on the semantic web; multimedia understanding through media analysis, annotation, and adaptation; multimodal data fusion; and multimodality ontology for multimedia. He has published over 100 refereed research papers.

Ivor W. Tsang received the Ph.D. degree in computer science from the Hong Kong University of Science and Technology, Kowloon, Hong Kong, in 2007. He is currently an Assistant Professor with the School of Computer Engineering, Nanyang Technological University (NTU), Singapore. He is the Deputy Director of the Center for Computational Intelligence, NTU.

Dr. Tsang was the recipient of the Natural Science Award (Class II) in 2008, China, which recognized his contributions to kernel methods, the prestigious

IEEE Transactions on Neural Networks Outstanding 2004 Paper Award in 2006, and a number of best paper awards and honors from reputable international conferences, including the Best Student Paper Award at CVPR 2010, the Best Paper Award at ICTAI 2011, and the Best Poster Award Honorable Mention at ACML 2012, and so on. He was also the recipient of the Microsoft Fellowship 2005, and ECCV 2012 Outstanding Reviewer Award.