Image Co-segmentation via Saliency Co-fusion

Koteswar Rao Jerripothula, *Student Member, IEEE*, Jianfei Cai, *Senior Member, IEEE*,
and Junsong Yuan, *Senior Member, IEEE*

***Abstract*—Most existing high-performance co-segmentation algorithms are usually complex due to the way of co-labeling a set of images as well as the common need of fine-tuning few parameters for effective co-segmentation. In this paper, instead of following the conventional way of co-labeling multiple images, we propose to first exploit inter-image information through co-saliency, and then perform single-image segmentation on each individual image. To make the system robust and to avoid heavy dependence on one single saliency extraction method, we propose to apply multiple existing saliency extraction methods on each image to obtain diverse salient maps. Our major contribution lies in the proposed method that fuses the obtained diverse saliency maps by exploiting the inter-image information, which we call saliency co-fusion. Experiments on five benchmark datasets with eight saliency extraction methods show that our saliency co-fusion-based approach achieves competitive performance even without parameter fine-tuning when compared with the state-of-the-art methods.**

***Index Terms*—Co-fusion, co-saliency, co-segmentation, fusion, saliency, segmentation.**

I. Introduction

IMAGE co-segmentation refers to the task of extracting common objects from a set of images, which is very useful for many vision and multimedia applications such as object-based image retrieval, image classification, and object recognition. It can be considered as one type of weakly supervised segmentation methods, which makes use of the weak prior that there exist common objects across different images in the set. This is quite different from single image segmentation. The existing single image object-level segmentation methods can only exploit either the prior from human supervision, which requires human interactions such as GrabCut, or the prior from single image-based visual saliency, which might fail at complex images with cluttered background or non-salient foreground. In contrast, image co-segmentation goes beyond single image segmentation in the sense that it can exploit not only the intra-image priors, but also the inter-image priors. Furthermore, it also brings in the new challenges of how to find the right inter-image priors and how to make use of them.

The concept of co-segmentation was first introduced in [1], which used histogram matching to simultaneously segment out the common object from a pair of images. Since then, many co-segmentation algorithms have been proposed in the literature, ranging from early image pair co-segmentation [2], [3], multiple image co-segmentation [4]–[7], interactive image co-segmentation [8]–[10] to the recent multiple objects co-segmentation [11]–[14], multiple group co-segmentation [15], noisy image set co-segmentation [16], large-scale co-segmentation [17], [18], shape alignment targeted co-segmentation [19] and evaluation criteria driven co-segmentation [20].

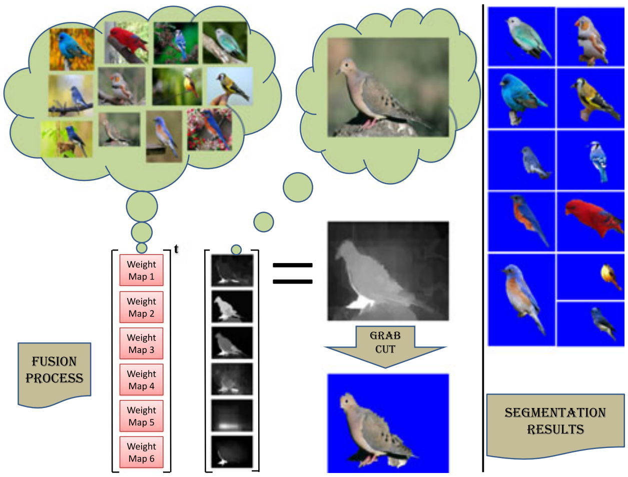

Fig. 1. Fusion of multiple saliency maps of an image generated by different saliency extraction methods to enhance the common foreground object while suppressing background saliency. The fusion process is essentially a weighted summation of different saliency maps at superpixel level.

Despite the great progress made by the existing co-segmentation algorithms, they still have some major limitations. First, most of the state-of-the-art co-segmentation algorithms require fine-tuning of quite a few parameters and the co-labeling of multiple images simultaneously, which are very complex and time-consuming, especially for large diverse datasets. Second, as seen in the existing works [16], [21], co-segmenting images might not perform better than single image segmentation for some datasets. This might be due to the additional energy term commonly used to enforce inter-image consistency, which often results into unsmooth segmentations in individual images.

In this paper, we focus on binary image co-segmentation, i.e. extracting a common foreground from a given image set. Instead of following the conventional way of co-labeling multiple images, we aim to exploit inter-image information through co-saliency, and then perform single-image segmentation on each individual image. Moreover, to make the system robust and avoid heavy dependence on one single saliency extraction method for generating co-saliency, we propose to apply multiple saliency extraction methods on each image. Eventually, an

Manuscript received July 31, 2015; revised November 24, 2015, February 11, 2016, March 28, 2016, and April 27, 2016; accepted May 19, 2016. Date of publication June 02, 2016; date of current version August 12, 2016. This work was supported in part by the Singapore MoE AcRF Tier-1 under Grant RG138/14. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Chengcui Zhang.

The authors are with the Nanyang Technological University, Singapore 639798 (e-mail: koteswar001@e.ntu.edu.sg; asjfcai@ntu.edu.sg; jsyuan@ntu.edu.sg).

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

Digital Object Identifier 10.1109/TMM.2016.2576283

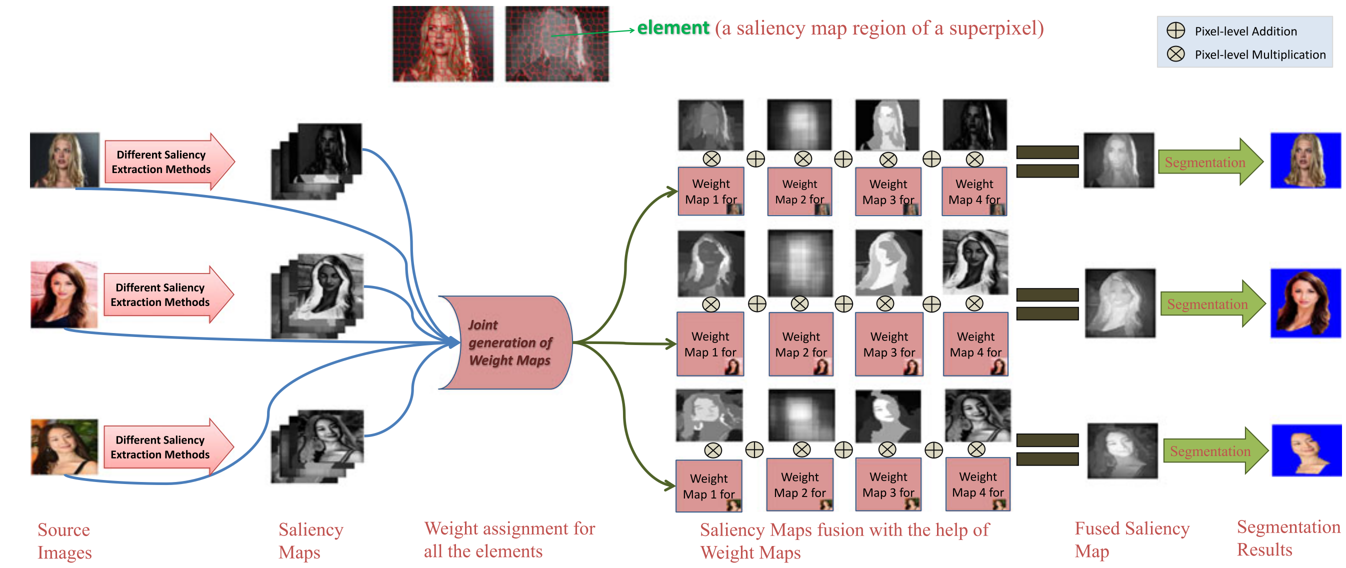

Fig. 2. Flowchart of the proposed saliency co-fusion-based image co-segmentation where multiple images are used to generate weight maps for fusing different saliency maps of images to extract a common foreground. Element, the basic processing unit of our process, is defined as a saliency map region of a superpixel.

TABLE I
NOTATIONS

Symbols	Meanings	References	Domains
$\mathcal{I}$	Imageset	Section III-A	
N and M	Number of images in $\mathcal{I}$ and number of saliency maps for each image	Section III-A	$\mathbb{R}$
n, k and m	Image, superpixel, saliency map indexes	Section III-A	$\mathbb{R}$
I^n	*n*th image	Section III-A	$\mathcal{I}$
H^n	Gaussian weight mask for *n*th image	Eqtion (11)	
$\mathcal{P}^n$ and $\mathcal{B}^n$	Superpixel set and saliency map set for *n*th image	Section III-A	
P_k^n	*k*th super-pixel of *n*th image	Section III-A	$\mathcal{P}^n$
B_m^n	*m*th saliency map of *n*th image	Section III-A	$\mathcal{B}^n$
J^n	Final fused saliency map of *n*th image	Equation (2)	
e	Elements	Section III-A	
N_e	Total number of elements	Section III-A	$\mathbb{R}$
$\mathbf{z}$	Weight vector for elements	Section III-A	$\mathbb{R}^{N_e \times 1}$
D	Prior term coefficient vector	Equations (1) and (6)	$\mathbb{R}^{N_e \times 1}$
G	Smoothness term coefficient matrix	Equations (1) and (14)	$\mathbb{R}^{N_e \times N_e}$
λ	Balancing parameter	Equation (1)	$\mathbb{R}$
d	Number of feature dimensions of an element	Section III-B	$\mathbb{R}$
X_f and X_b	Foreground and background feature matrices	Section III-B	$\mathbb{R}^{N_e \times d}$
X	Total feature matrix	Section III-B	$\mathbb{R}^{N_e \times 2d}$
S_f, S_b and S	Foreground, background and total similarity matrices	Equations (3)–(5)	$\mathbb{R}^{N_e \times N_e}$
D_s, D_f and D_c	Saliency, foreground/background and centerness cue vectors	(8), (10) and (13)	$\mathbb{R}^{N_e \times 1}$
T and F	Average and recommended saliency vector	Section III-C and Equation (7)	$\mathbb{R}^{N_e \times 1}$
C	Spatial weight vector	Equation (12)	$\mathbb{R}^{N_e \times 1}$
u and v	Element vector indexes	Section III-A	$\mathbb{R}$
$R_u(v)$	Saliency punishment recommended by element v to u	Equation (9)	$\mathbb{R}$
V and Q	Neighborhood matrix and diagonal matrix composed of its row sums	Equations (14) and (15)	$\mathbb{R}^{N_e \times N_e}$
γ	Normalization parameter	Equations (3)–(5), and (15)	$\mathbb{R}$
θ	Similarity threshold	Section III-B, Equations (7) and (10)	$\mathbb{R}^{N_e \times 1}$
τ and ϕ^n	Parameters for final segmentation	Equation (16)	$\mathbb{R}$

enhanced saliency map is generated for each image by fusing its various saliency maps via weighted summation at superpixel level, where the weights are optimized by exploiting inter-image information, as shown in Fig. 1. We call the proposed method saliency co-fusion, whose objectives include: 1) boosting the saliency of common foreground regions; and 2) suppressing the saliency of background regions.

Fig. 2 illustrates the process flow of the proposed saliency co-fusion based image co-segmentation. The key component lies in the developed saliency co-fusion process, which is performed at the superpixel level. Particularly, we define each saliency map region (produced by one saliency detection method) of one superpixel as an element (see Fig. 2 top), and give a weight for each element. We formulate the weight selection as an energy minimization problem, where we incorporate saliency recommendations from similar elements, foreground/background priors through similar element voting, and neighbor smoothness constraints. Finally, the fused saliency for a superpixel is just a weighted summation of all the saliency maps of the superpixel. Experimental results show that our saliency co-fusion

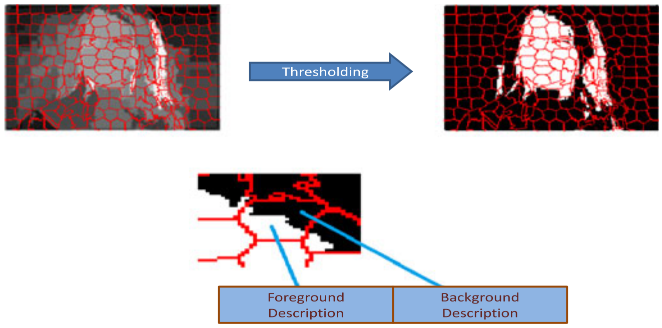

Fig. 3. Feature description. Each element gets divided into foreground and background regions after global thresholding and two sets of features are extracted from these two regions separately, one for foreground and the other for background.

based co-segmentation achieves competitive performance even without fine-tuning the parameters, i.e., at default setting, compared with the state-of-the-art co-segmentation algorithms. In addition, our by-product, the fused saliency map, exhibits some attractive properties, which could be useful for many other applications.

II. Related Work

Our method is closely related to co-segmentation and co-saliency research.

Co-segmentation: Many co-segmentation algorithms have been proposed in the literature. Early approaches [1]–[3] focused on segmenting a pair of images containing one common object. It was later extended to deal with multiple images containing one common object with more effective or more efficient models enforcing inter-image consistency [4], [5], [22]–[25]. However, there are also some algorithms being designed for segmenting multiple common foregrounds from a given image set [11]–[14], where the best performers make use of supervised information. On the contrary, our method is purely an unsupervised approach. A few interactive co-segmentation approaches [8]–[10] have also been proposed, where users can give scribbles for one or a small number of the images. Thus, the extracted prior information is then used to influence the segmentation of the entire image set. Our method does not require such human intervention. Recently, [16] applied dense SIFT matching to discover common objects, and co-segment them out from noisy image dataset, where some images do not contain common objects. They tried to enforce inter-image consistency strongly by developing matching based prior, so as to exclude noise image from participating in the co-segmentation process. In [19], co-segmentation was combined with co-sketch for effective co-segmentation by sharing shape templates. In [26], co-segmentation problem was addressed by establishing consistent functional maps across images in a reduced functional space, which requires training. Another interesting work [20], which reports state-of-the-art performance, employed region-level matching. Also, it determined a good co-segment by checking whether it can be well composed from other co-segments. Most of these methods focused on pixel level co-labeling whereas we focus on saliency co-fusion.

Just like we use multiple saliency maps, there are also some methods that use multiple segmentation proposals to perform semantic segmentation. For example, [27] made use of multiple segmentation proposals of an image to come up with several compositions and eventually produce semantic segmentation by searching for high-scoring maximal weighted cliques. Later, [21] extended the idea of using multiple segmentation proposals to the object co-segmentation problem and demonstrated better results than classical co-segmentation algorithms. In this research, in contrast to pre-segmenting and then selecting segmentation proposals, we propose to fuse multiple saliency maps to arrive at an enhanced saliency map and then carry out segmentation.

Co-saliency: Co-saliency typically refers to the common saliency existing in a set of images containing similar objects. The term "co-saliency" was first coined in [28], in the sense of what is unique in a set of similar images, and the concept was later linked to extract common saliency, which is very useful for many practical applications [29], [30]. For example, co-saliency object priors have been efficiently used for co-segmentation in [31]. Recently, a cluster based co-saliency method using various cues was proposed in [32], which learns the global correspondence and obtains cluster saliency quite well. It represents state-of-the-art method due to its simplicity, effectiveness, and efficiency. However, the co-saliency method [32] is mainly designed for images of the same object captured at different viewpoints or instances. It cannot well handle image sets with huge intra-class variation. Another recent work [33] fused saliency maps from different images *via* warping technique and it is able to handle the intra-class variation. However, most of these methods only use a single saliency map which may not be accurate always.

III. Saliency Co-Fusion

In this section, we first formulate our saliency co-fusion problem. Then we give a detailed description of individual terms as well as implementation details.

A. Problem Formulation

Considering a set of N images $\mathcal{I} = \{I^1, I^2, ..., I^N\}$, denote $\mathcal{B}^n = \{B_1^n, B_2^n, ..., B_M^n\}$ the set of M saliency maps (normalized to range 0–1) for image I^n obtained using M different existing saliency extraction methods. Also, denote $\mathcal{P}^n = \{P_1^n, P_2^n, ..., P_{|\mathcal{P}^n|}^n\}$ the set of superpixels in image I^n obtained using [34]. Defining a saliency map region of superpixel as an element e, which is the basic processing unit in our method, we have total $N_e = \sum_{n=1}^{N} M|\mathcal{P}^n|$ elements. Let $z(n,k,m)$ denote the associated weight for element $e(n,k,m)$ that belongs to image n, superpixel k, and saliency map m. The weight maps depicted in Figs. 1 and 2 are basically constructed using these associated weights.

We stack all the weights into a vector $\mathbf{z} = [z_1, z_2, \ldots, z_{N_e}]^t$ for simplicity and use u or v as the element indexes for referencing purposes. We mix the usage of the element vector index

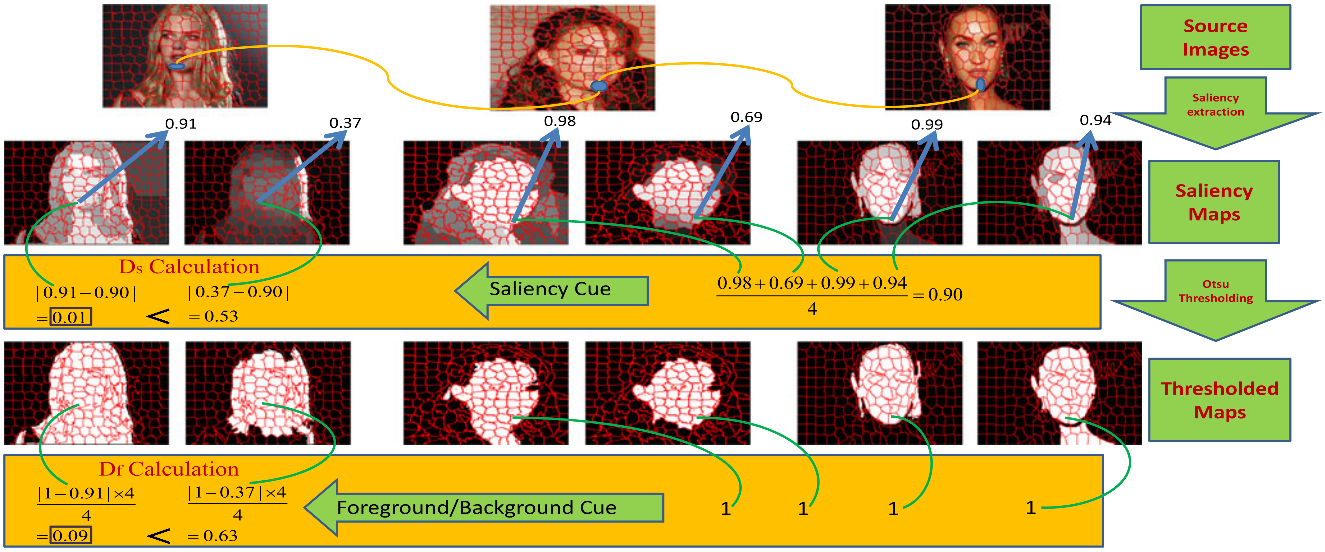

Fig. 4. Illustration of how similar elements from other images help in determining better elements via D_s (saliency cue) and D_f (foreground/background cue) calculations in the first image. Note that the numerical value that an element is pointing to is the average saliency value of the element. D_s signifies how close the saliency value of an element is to the recommended saliency value by its similar elements, whereas D_f signifies the average punishment of an element for deviating from the foreground/background recommendations from each of its similar elements. The lesser the D_s and D_f are for an element, the higher weight the element will get. For example, the element covering the chin area in the first saliency map is considered a better one than that in the second saliency map because of having lower values for both of the cues.

TABLE II
EVALUATION ON MSRC DATASET USING JACCARD SIMILARITY METRIC WHERE INDIVIDUAL SALIENCY MAPS AND FUSED SALIENCY MAPS ARE SEGMENTED USING OTSU'S METHOD

Class	[39]	[40]	[41]	[42]	[43]	[44]	[45]	[46]	AVG	MAX	Ours
Car	0.541	0.466	0.381	0.469	0.510	0.598	0.507	0.629	0.660	0.666	**0.696**
Sheep	0.745	0.699	0.736	0.612	0.776	0.615	0.697	0.744	0.793	0779	**0.810**
Cow	0.670	0.670	0.673	0.603	0.734	0.658	0.653	0.736	0.742	0.729	**0.794**
Flower	0.705	0.679	0.556	0.627	0.694	0.625	0.641	0.688	0.721	0.726	**0.768**
Cat	0.439	0.526	0.560	0.597	0.573	0.565	0.539	0.609	0.651	0.624	**0.714**
Sign	0.746	0.619	0.552	0.567	0.646	0.669	0.570	0.796	0.775	0.743	**0.812**
Tree	0.636	0.655	0.471	0.400	0.632	0.601	0.561	0.606	0.681	0.669	**0.738**
House	0.586	0.613	0.486	0.389	0.528	0.630	0.450	0.640	0.670	0.669	**0.712**
Dog	0.527	0.559	0.503	0.520	0.469	0.452	0.503	0.627	0.628	0.582	**0.643**
Bird	0.529	0.590	0.573	0.535	0.590	0.459	0.583	0.611	0.644	0.589	**0.662**
Bike	0.377	0.416	0.297	0.3827	0.436	0.453	0.463	0.420	0.488	0.473	**0.548**
Chair	0.546	0.588	0.566	0.474	0.595	0.530	0.496	0.563	**0.638**	0.588	**0.638**
Face	0.515	0.411	0.395	**0.582**	0.463	0.446	0.367	0.548	0.565	0.567	0.571
Plane	0.420	0.437	0.399	0.297	0.505	0.505	0.475	**0.542**	0.535	0.469	0.518
Avg	0.570	0.566	0.510	0.504	0.582	0.558	0.536	0.626	0.656	0.634	**0.688**

with its corresponding matrix index (n, k, m) since one can be converted to the other easily. Table I summarizes the major notations used throughout the paper.

Our goal is to find the optimal weight for each of the elements in order to jointly fuse various saliency maps of similar images at superpixel level such that common foreground saliency gets boosted up and background saliency is suppressed in final fused saliency maps. In particular, we treat saliency co-fusion as a weight selection problem. On one hand, we want to give higher weights to elements with higher confidence. On the other hand, we want to have certain consistency in the weight selection among neighboring elements. Considering the constraint that the resultant fused saliency map values should occur in the range [0, 1], we formulate our task as a quadratic programming problem

$$\begin{aligned} \min_{\mathbf{z}} \quad & D^t\mathbf{z} + \lambda \mathbf{z}^t G\mathbf{z} \\ \text{s.t.} \quad & 0 \le z_u \le 1,\ \forall u \in [1, N_e], \\ & \sum_{m=1}^{M} z(n,k,m) = 1,\ \forall I^n \in \mathcal{I}, P_k^n \in \mathcal{P}^n \end{aligned} \tag{1}$$

where there are two terms traded off by a balancing parameter λ. The first term ($D^t\mathbf{z}$) is a prior term to enforce global commonness and co-saliency, where the prior term coefficient vector $D \in \mathbb{R}^{N_e \times 1}$. The second term ($\mathbf{z}^t G\mathbf{z}$) is a pairwise smoothness term to encourage neighborhood elements to take similar weights, where the smoothness term coefficient matrix

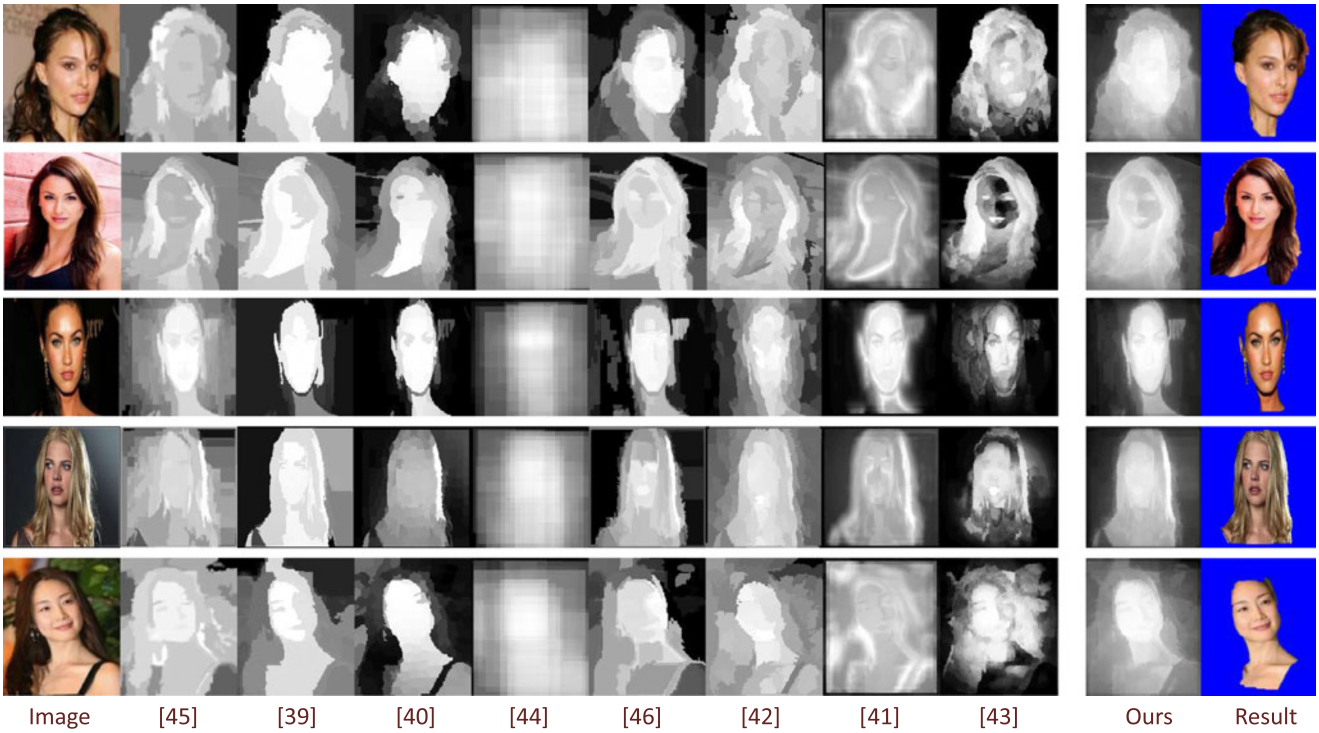

Fig. 5. Examples to illustrate the advantages of the fused saliency maps over the input saliency maps.

$G \in \mathbb{R}^{N_e \times N_e}$. The constraints in (1) are there to ensure that individual weights range between 0 and 1, and the summation of all the weights for one superpixel is equal to one. Once $\mathbf{z}$ is determined by minimizing (1), the fused saliency map J^n for a pixel $p \in P_k^n$ can be simply computed as

$$J^n(p) = \sum_{m=1}^{M} z(n,k,m) \times B_m^n(p) \tag{2}$$

where B_m^n is the m^{th} saliency map for image I^n.

B. Feature Description and Similarity

Unlike other methods [16], [32], [33] where features for matching are extracted from images independent of saliency maps, we develop a saliency map based feature descriptor because our processing units are elements (defined as a saliency map region of a superpixel), instead of pixels or superpixels. We consider the fact that there is no uniformity among saliency maps obtained by different methods. For instance, some saliency maps are of high contrast, while others are of poor contrast. Some are bright, while others are dark. This can cause serious problems in the process if saliency values are directly taken as features. We tackle it by distinguishing potential foreground pixels from potential background pixels in an element using the classical Otsu's method as shown in Fig. 3. For each group (both the potential foreground group and the potential background group in the element), we construct a feature descriptor which consists of the average dense SIFT descriptor, and also the average color values in RGB, HSV, and Lab spaces. However, for each element, we have two feature descriptors with each having dimensions $d = 128 + 3 + 3 + 3 = 137$. We concatenate them as the feature descriptor for one element. In this way, different elements of the same superpixel obtain different feature descriptors, depending upon the foreground/background distributions in each element.

$X_f, X_b \in \mathbb{R}^{N_e \times d}$ and $X \in \mathbb{R}^{N_e \times 2d}$ denote the data matrices that stack the foreground descriptors, the background descriptors and the foreground background concatenated descriptors of all the elements as its rows, respectively. We construct similarity matrices S_f, S_b and S, all of $N_e \times N_e$ dimensions that record the potential foreground similarity, the potential background similarity, and the total similarity, respectively, between all the element pairs

$$S_f(u,v) = \exp\left(-\gamma \sum_{q=1}^{d} \frac{\left(X_f(u,q) - X_f(v,q)\right)^2}{X_f(u,q) + X_f(v,q)}\right) \tag{3}$$

$$S_b(u,v) = \exp\left(-\gamma \sum_{q=1}^{d} \frac{\left(X_b(u,q) - X_b(v,q)\right)^2}{X_b(u,q) + X_b(v,q)}\right) \tag{4}$$

$$S(u,v) = \exp\left(-\gamma \sum_{q=1}^{2d} \frac{\left(X(u,q) - X(v,q)\right)^2}{X(u,q) + X(v,q)}\right) \tag{5}$$

where γ is a parameter set to $\frac{1}{300}$.

Note that the potential foreground similarity is set to zero if all the pixels in the element belong to the background group and vice versa. If elements u and v belong to the same image, $S_f(u,v)$, $S_b(u,v)$, and $S(u,v)$ are all set to zero since we aim at exploiting similar elements from other images.

Based on the total similarity matrix S, similar elements for each element are identified if the corresponding similarity values are large than a similarity threshold θ (θ is set to 0.75). For one element, its similar elements provide recommendations via different cues, based on which we then derive the appropriate weight

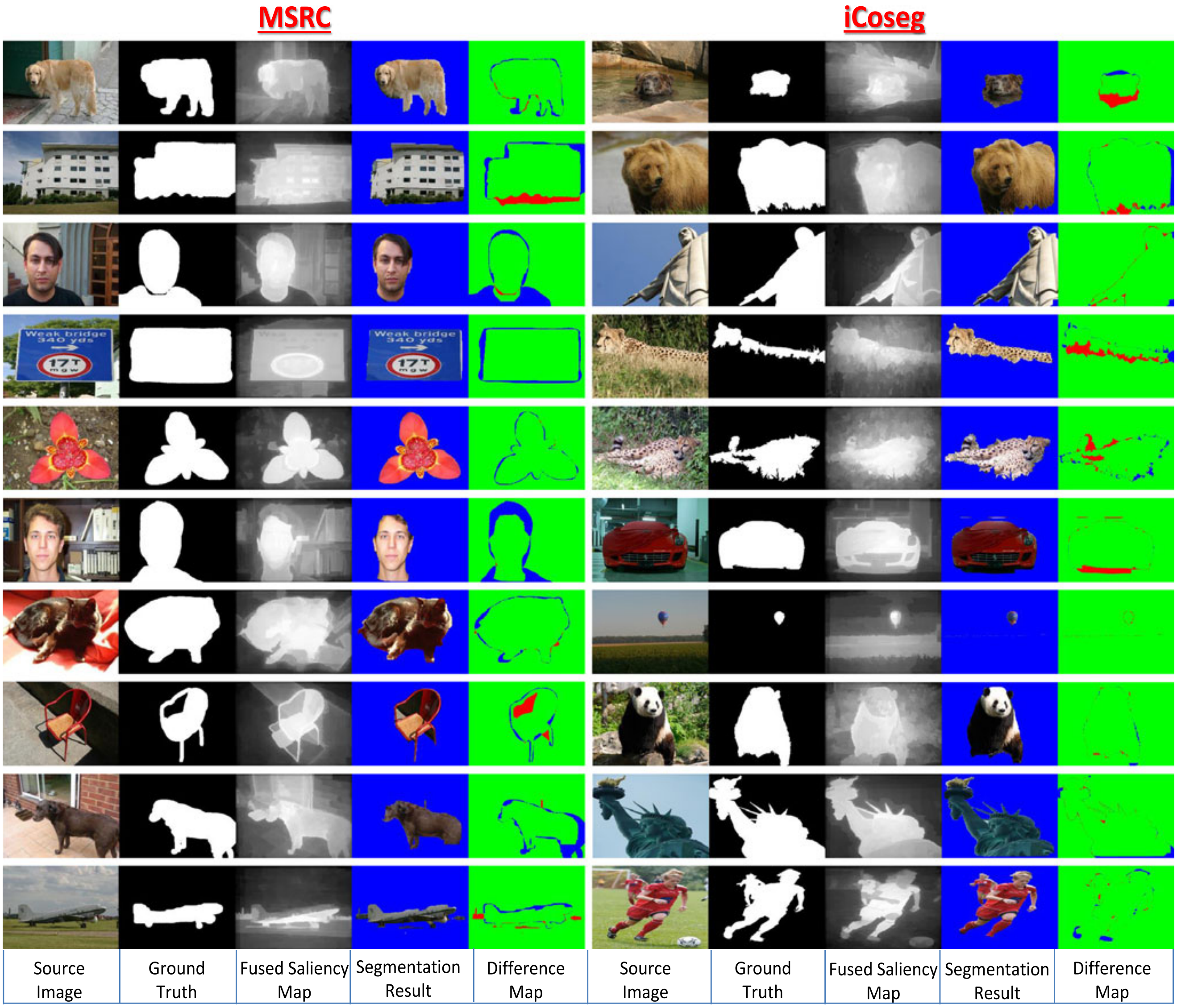

Fig. 6. Sample examples of ground-truth images, fused saliency maps, our segmentation results and the difference maps (our results minus the corresponding ground-truth images) on MSRC and iCoseg datasets. Note that for the difference maps, green, red, and blue correspond to 0, 1, −1, respectively.

for the considered element so as to encourage or discourage its role in the final fused saliency map. Details are elaborated below.

C. Prior Term

We define our prior term coefficient vector D in (1) as

$$D = D_s + D_f + D_c \tag{6}$$

which includes three cues: saliency cue (D_s) from similar elements, foreground/background prior cue (D_f) from similar elements and centerness cue (D_c) based on the spatial location of the element.

Saliency Cue: Following the idea of co-saliency or common saliency, we compare the average saliency of similar elements with the average saliency value of the considered element to decide whether the element should be emphasized or not (give high weight or not). Let $T = [T_1, T_2, ..., T_{N_e}]^t$ denote the vector where each entry is the average saliency value of an element. On the other hand for an element u, we compute the average saliency recommended by its similar elements as

$$F_u = \frac{\sum_{v=1}^{N_e} T_v \delta\big(S(u,v) > \theta\big)}{\sum_{v=1}^{N_e} \delta\big(S(u,v) > \theta\big)} \tag{7}$$

where $\delta\big(\cdot\big)$ is the indication function, equal to one if the condition $\big(\cdot\big)$ is true (otherwise 0), which is used to determine whether element v is a similar one or not. Let $F = [F_1, F_2, ..., F_{N_e}]^t$ be the vector comprising of the recommended average saliency values of elements. We then simply define the saliency cue as

$$D_s = |F - T|. \tag{8}$$

Essentially, (8) suggests that if $T(u)$ is very different from $F(u)$, then the corresponding weight z_u is encouraged to be small by (1). Fig. 4 illustrates how similar elements from other images help to determine better elements.

TABLE III
OVERALL JACCARD SIMILARITY (JACC.) AND ACCURACY (ACC.) RESULTS ON DIFFERENT DATASETS USING OUR METHODS THAT RESPECTIVELY INCORPORATE OTSU'S METHOD AND GRABCUT METHOD WITH THE DEFAULT SETTING [$\tau = 0.75$] FOR SEGMENTATION

	Otsu's method		GrabCut	
	Jacc.	Acc.	Jacc.	Acc.
MSRC	0.69	86.7	0.70	87.9
iCoseg	0.65	87.0	0.70	89.7
Coseg-Rep	0.71	89.5	0.76	92.7
Car	0.70	85.3	0.69	86.0
Horse	0.49	78.5	0.55	83.9
Airplane	0.52	82.6	0.56	86.8
FlickrMFC	0.60	83.5	0.67	87.0

TABLE IV
PERFORMANCE RESULTS BY VARYING REGION-SIZE PARAMETER OF SLIC [34] ON MSRC DATASET

	20	40	60	80	100
Jacc.	0.6878	0.6877	0.6870	0.6869	0.6865
Acc.	86.31	86.30	86.27	86.26	86.25

TABLE V
COMPARISON ON COSEG-REP DATASET USING OVERALL VALUES OF JACCARD SIMILARITY (JACC.) AND ACCURACY (ACC.)

	Jacc.	Acc.
Cosegmentaton Cosketch [19]	0.67	90.2
Geometric Mean Saliency [33]	0.73	92.2
Ours (tuned)	**0.77**	**93.4**

TABLE VI
COMPARISON WITH STATE-OF-THE-ART METHODS ON INTERNET IMAGES DATASET USING OVERALL VALUES OF JACCARD SIMILARITY (JACC.) AND ACCURACY (ACC.)

	Car		Horse		Airplane	
	Jacc.	Acc.	Jacc.	Acc.	Jacc.	Acc.
[4] (reported in [16])	0.37	58.7	0.30	63.8	0.15	49.2
[50] (reported in [16])	0.35	59.2	0.29	64.2	0.12	47.5
[16]	0.63	83.4	0.54	83.7	0.56	86.1
Ours (default)	0.69	86.0	0.55	83.9	0.56	86.8
Ours (tuned)	**0.71**	**88.0**	**0.60**	**88.3**	**0.61**	**90.5**

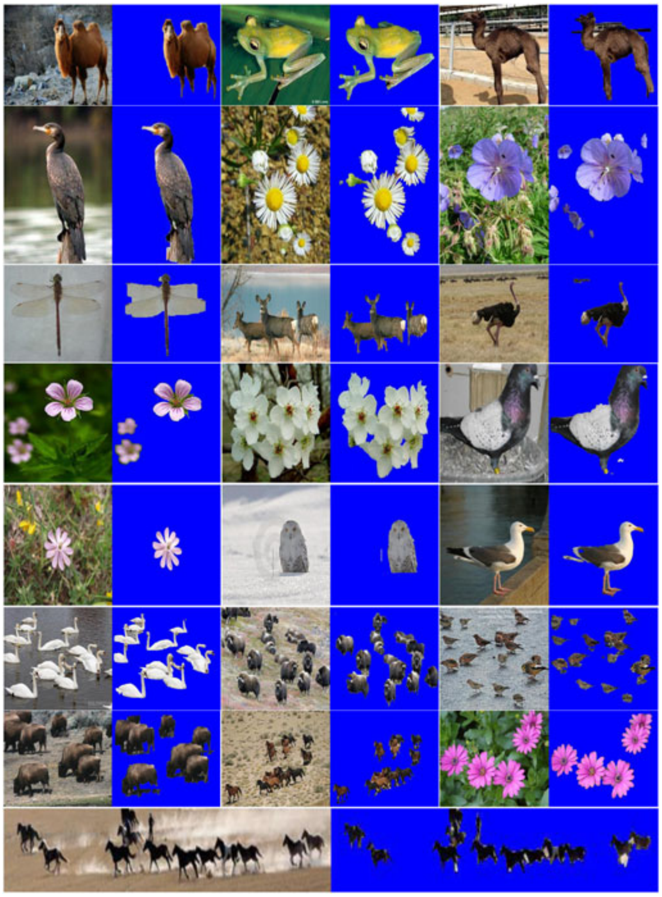

Fig. 7. Sample segmentation results on Coseg-Rep dataset.

Foreground/Background Cue: Another cue similar elements can provide is to recommend the given element to be foreground or background. For an element u and one of its similar elements v, if their foreground feature descriptors are more similar than the background descriptors, v recommends foreground with a saliency punishment of $(1 - T(u))$ to u; otherwise, it recommends background with a punishment of $(T(u) - 0)$, i.e.

$$\begin{aligned} R_u(v) &= 1 - T(u), \text{ if } \quad S_f(u,v) > S_b(u,v) \\ R_u(v) &= T(u) - 0, \text{ if } \quad S_f(u,v) < S_b(u,v) \end{aligned} \tag{9}$$

where $R_u(v)$ denotes the saliency punishment recommended by v to u. Considering all the similar elements, we define foreground/background cue D_f for an element u as

$$D_f(u) = \frac{\sum_{v=1}^{N_e} \delta\big(S(u,v) > \theta\big) R_u(v)}{\sum_{v=1}^{N_e} \delta\big(S(u,v) > \theta\big)} \tag{10}$$

where $\delta(\,\cdot\,)$ is the indication function, equal to one if the condition $(\,\cdot\,)$ is true (otherwise 0), so as to include only similar elements. Fig. 4 also illustrates how similar elements from other images provide the foreground/background cue.

Centerness Cue: In addition to the above mentioned saliency and foreground/background cues, we also take advantage of the general observation that objects are often located at the center, and such central bias is quite prevalent in several benchmark datasets as pointed out in [35]. Therefore as an extra measure, saliency maps that emphasize center regions are encouraged to be given higher weights at central regions. To account for central bias, a spatial weight mask for each image is created using normalized Gaussian function which is centered at the image center. Specifically, for a pixel p in I^n (of size $width_n \times height_n$) with coordinates (x, y) and with its origin at the image center, the central weight mask is defined as

$$H^n(p) = \exp\left(-\frac{x^2}{0.2 \times width_n^2} - \frac{y^2}{0.2 \times height_n^2}\right). \tag{11}$$

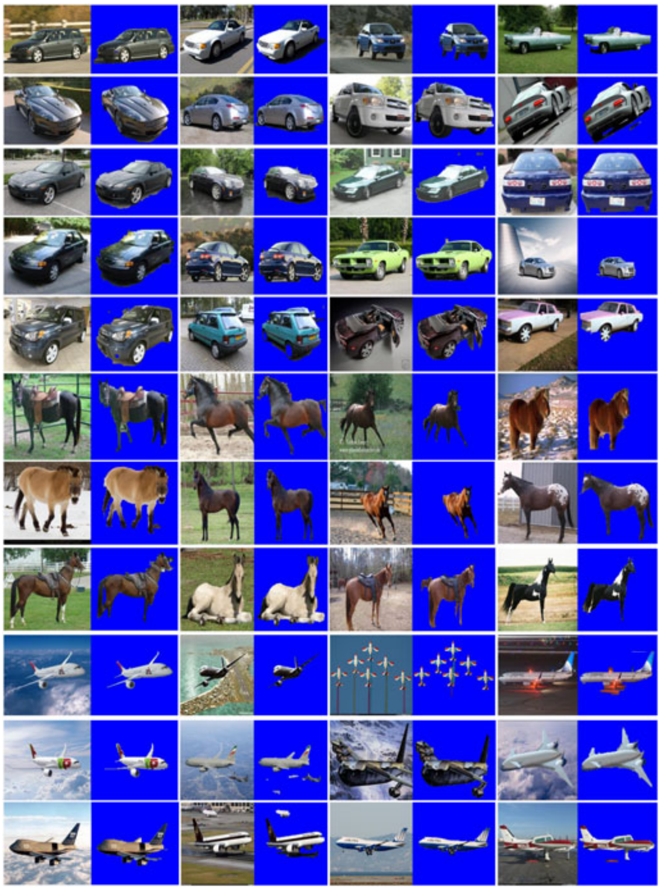

Fig. 8. Sample segmentation results on Internet images dataset containing three categories: (i) car, (ii) horses, and (iii) airplane.

For an element u or $e(n,k,m)$, its central bias is calculated by averaging the spatial weights of all its pixels, i.e.

$$C_u = \frac{\sum_{p\in P_k^n} H^n(p)}{\sum_{p\in P_k^n} 1}. \tag{12}$$

Let $C = [C_1, C_2, ..., C_{N_e}]^t$ denote the vector consisting of the central bias weights of all the elements. Thus, we now define the centerness cue D_c for an element u as

$$D_c(u) = C(u) \times |C(u) - T(u)| \tag{13}$$

which essentially measures how the saliency of an element deviates from its central bias weight. Central bias weight is also multiplied so that influence of this deviation in minimizing (1) depends upon the spatial location of the element.

Note that our centerness cue is different from other methods like [32], which deliberately emphasize the center regardless of whether an object is present or not in the center. On the contrary, our method emphasizes the center only if a salient object is present in the center. Our centerness cue provides additional support when the saliency and foreground/background cues fail to recommend something substantial because of lack of support from other images due to too much intra-class variation or pose differences. In such case, if there is a salient object at the center, it will be supported by the centerness cue.

TABLE VII
COMPARISON WITH STATE-OF-THE-ART METHODS ON MSRC AND ICOSEG DATASETS USING OVERALL VALUES OF JACCARD SIMILARITY (JACC.) AND ACCURACY (ACC.)

	MSRC		iCoseg	
	Jacc.	Acc.	Jacc.	Acc.
Discriminative [4]	0.45	70.8	0.39	61.0
Multi-Class [50]	0.51	73.6	0.43	70.2
Object Discovery [16]	0.68	87.7	0.69	89.8
Geometric Mean Saliency [33]	0.70	88.4	0.72	91.6
Composition [20]	**0.73**	**89.2**	**0.73**	**92.8**
Ours (tuned)	0.71	88.7	0.72	91.9

TABLE VIII
COMPARISON ON FLICKRMFC DATASET USING OVERALL JACCARD SIMILARITY (JACC.) VALUE. (U) MEANS UNSUPERVISED METHOD AND (S) MEANS SUPERVISED METHOD

Methods	Jacc.
Multiple Foreground Cosegementation (U) [14]	0.322
Multiple Foreground Cosegementation (S) [14]	0.482
Discriminative Clustering (U) [50]	0.414
Directed Graph Clustering (U) [11]	0.547
Graph Transduction (S) [13]	0.626
w/o NON RIGID Mapping (U) [12]	0.589
with NON RIGID Mapping (S) [12]	0.647
Ours (U) (default)	0.667
Ours (U) (tuned)	**0.684**

D. Smoothness Term

Since in our prior term we have made discrete conditions using θ to select similar elements, there is a certain possibility of inconsistencies in weight distribution. A smoothness term is necessary to curb inconsistencies in weight distribution among neighbor elements. Here we define neighbor elements as those which are similar in not only the feature space but also the saliency space. If a pair of elements have very similar saliency and are very close in the feature space as well, they should be encouraged to have similar weights. Thus, the smoothness term $\mathbf{z}^t G\mathbf{z}$ is introduced to ensure that these neighbor elements in both feature space and saliency space take similar weights. However, we use the conventional normalized Laplacian matrix for defining smoothness term coefficient G in (1), similar to [36], i.e.

$$G = A - Q^{-\frac{1}{2}} V Q^{\frac{1}{2}} \tag{14}$$

where A is the identity matrix, V is neighborhood matrix, and Q is the diagonal matrix composed of row sums of matrix V. In addition, different from the similarity matrix S defined in (5), V takes into account similarity in both feature space and saliency space, i.e.

$$V(u,v) = \exp\left(-\gamma \frac{\sum_{q=1}^{2d} \frac{\left(X(u,q)-X(v,q)\right)^2}{X(u,q)+X(v,q)}}{2d} - |T(u) - T(v)|\right) \tag{15}$$

where γ is a normalization parameter set to $\frac{1}{300}$, which is the same as that in (3)–(5).

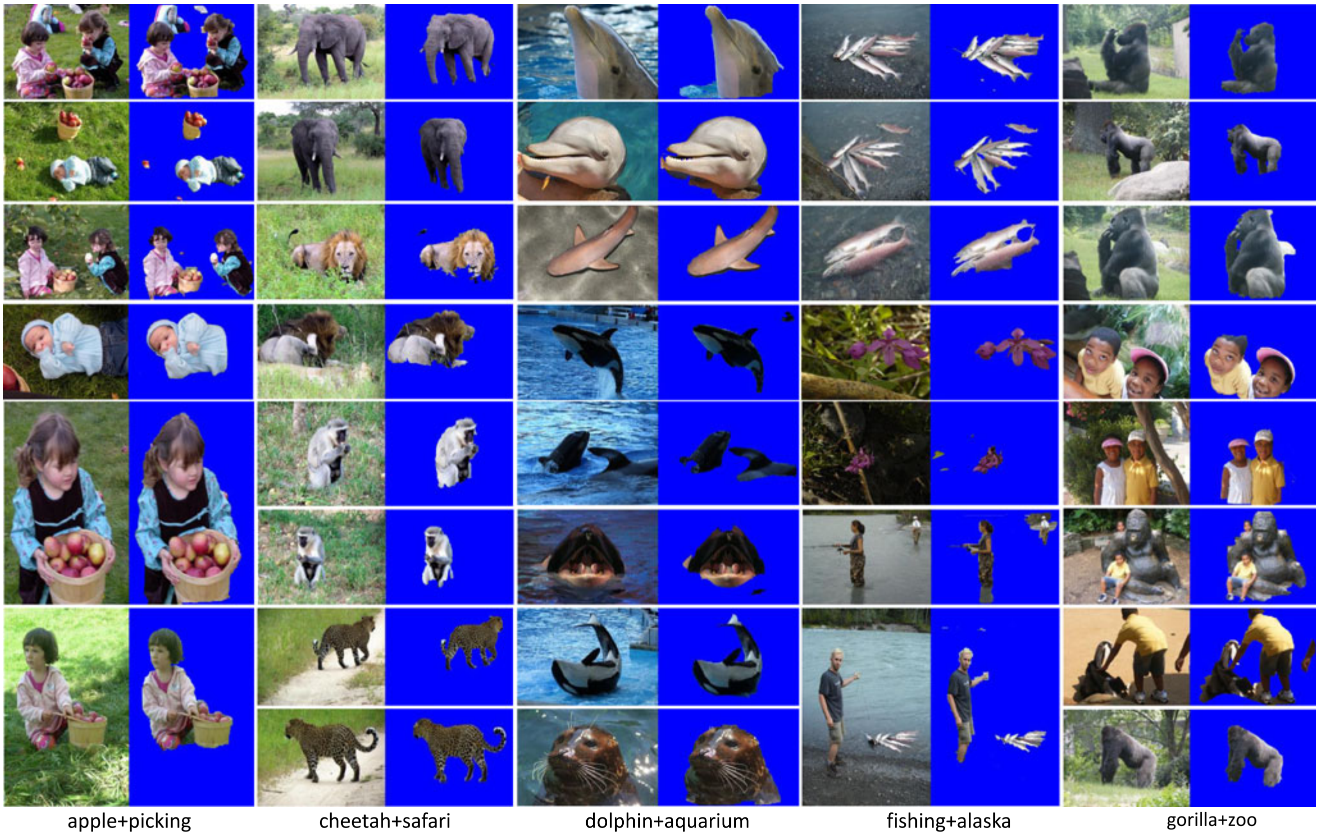

Fig. 9. Sample segmentation results on FlickrMFC dataset.

E. Implementation Details

For optimization, since G is positive semi-definite and the constraints are linear, the objective function defined in (1) is essentially a quadratic programming problem, which is solved by the interior-point convex algorithm provided in Matlab.

Once the fused saliency map is available, different single-image segmentation algorithms can be applied for segmentation. In this research, we adopt two segmentation methods as two variations. One is the classical Otsu's method, which is an optimal threshold based method. The other one is GrabCut algorithm [37] with some modification. Specifically, by noticing the final fused saliency map containing certain boundary information, following [17], we modify the GrabCut energy equation and add another localization potential to ensure that segmentation is guided not only by color, but also by the location prescribed by the object prior contained in the fused saliency map. The foreground (FG) and the background (BG) seed locations are determined by

$$p \in \begin{cases} FG, & \text{if } J^n(p) > \tau \\ BG, & \text{if } J^n(p) < \phi^n \end{cases} \qquad (16)$$

where ϕ^n is a global threshold value automatically determined by the classical Otsu's method and τ is a parameter (by default $\tau = 0.75$). It should be noted that other single-image segmentation methods such as [38] can also be used for the final segmentation.

IV. Experimental Results

We conducted extensive experiments on five existing benchmark co-segmentation datasets (MSRC [47], iCoseg [8], Coseg-Rep [19], Internet images dataset [16], and FlickrMFC dataset [14]). As mentioned in the introduction, the existing methods often require fine tuning of quite a few parameters. In order to demonstrate the effectiveness of our method, we make two types of settings in our experiments: 1) default parameter settings for all the categories in the datasets and 2) tuning parameter τ over categories for a fair comparison with other methods. Following the literature, we adopted two evaluation metrics: (i) Jaccard Similarity (Jacc.) [48] and (ii) Accuracy (Acc.). Denote A_p^f, A_p^b, A_g^f and A_g^b as proposed foreground pixels set, proposed background pixels set, groundtruth foreground pixels set and groundtruth background pixels set, respectively. Here, Jaccard Similarity is defined as the size of intersection divided by the size of union of the proposed and groundtruth foreground pixels sets, i.e. $\frac{|A_p^f \cap A_g^f|}{|A_p^f \cup A_g^f|}$. And Accuracy is defined as the percentage of pixels that have same labels in both the proposed and groundtruth masks, i.e.

$$\frac{|A_p^f \cap A_g^f| + |A_p^b \cap A_g^b|}{|A_g^f \cup A_g^b|} \times 100.$$

TABLE IX
CLASS-WISE JACCARD SIMILARITY PERFORMANCE ON MSRC DATASET

	Car	Sheep	Cow	Flower	Cat	Sign	Tree	House	Dog	Bird	Bike	Chair	Face	Plane
[16]	0.667	0.789	0.794	0.714	0.662	0.823	0.699	0.727	0.675	0.673	0.541	0.622	0.583	0.567
[33]	0.704	0.799	0.801	0.723	**0.760**	0.839	**0.772**	0.764	0.683	0.628	0.462	0.650	0.604	0.543
[20]	0.710	**0.850**	**0.880**	**0.790**	0.700	**0.850**	0.760	**0.840**	0.690	**0.680**	**0.580**	**0.730**	**0.630**	**0.580**
ours	**0.713**	0.811	0.812	0.770	0.734	0.831	0.769	0.752	**0.699**	0.665	0.544	0.671	0.608	0.552

TABLE X
CLASS-WISE JACCARD SIMILARITY PERFORMANCE ON ICOSEG DATASET

	Base ball	Bear2	Brown bear	Cheetah	Christ	Elephant	Ferrari	Goose	Gymna-stic1	Gymna-stic2	Gymna-stic3	Helico-pter	Hotba-lloon	Kendo	Kendo2
[16]	0.657	0.653	0.736	0.697	0.770	0.688	**0.724**	0.742	0.948	0.839	0.896	0.803	0.657	0.778	0.826
[33]	**0.756**	0.701	0.662	0.754	0.795	0.735	0.703	0.773	0.910	**0.897**	**0.911**	0.766	0.763	0.862	0.893
[20]	0.610	**0.720**	**0.920**	0.670	**0.870**	0.670	0.680	**0.870**	0.970	0.820	0.900	**0.820**	**0.880**	0.890	**0.960**
ours	0.703	0.675	0.725	**0.780**	0.757	**0.799**	0.708	0503	**0.976**	0.831	0.892	0.803	0.802	**0.896**	0.921
	Liver pool	Monk	Panda 1	Panda 2	Pyramid	Skate	Skate 2	Skate 3	Statue	Stone-henge	Taj mahal	Track& field	Wind mill	Women soccer	Women soccer2
[16]	**0.541**	0.681	0.759	0.625	0.611	0.735	**0.910**	0.449	0.799	0.595	0.460	0.519	0.492	0.661	0.530
[33]	0.512	0.688	**0.806**	**0.718**	**0.686**	0.737	0.866	0.297	0.813	0.714	0.587	0.632	0.316	0.657	**0.538**
[20]	0.470	**0.800**	0.700	0.550	0.580	**0.910**	0.690	0.160	0.770	**0.910**	**0.840**	**0.660**	**0.570**	0.660	0.460
ours	0.470	0.683	0.722	0.614	0.595	0.769	0.900	**0.491**	**0.863**	0.781	0.516	0.595	0.531	**0.699**	0.526

We use eight saliency extraction methods [39]–[46] to generate various saliency maps as the input to our method. In the following subsections, we first briefly introduce the datasets used, followed by individual experiments, discussions and comparisons.

A. Datasets

MSRC dataset contains 14 categories with 418 images in total. Coseg-Rep dataset contains 23 categories and 572 images in total, where there is a special category named "Repetitive" that has several instances of the same type of object within one image (e.g., an image containing multiple horses). Both of the datasets exhibit intra-class variation. As a result, we do not use the color feature for matching the elements as it will be unreliable.

iCoseg dataset contains 38 categories with 643 images in total. For a fair comparison with the existing methods [16], [20], we use the same part of the dataset in our experiments. This includes 30 categories and a total of 530 images. Flickr MFC dataset contains multiple common objects that might not appear in every image. It has 14 categories and 263 images in total. For these two datasets, since the same objects appear frequently across the images, we include the color features in our method. Also, Internet images dataset created by [16] contains three categories: Airplane, Car, and Horse, with 4347, 6381 and 4542 images respectively, where only some of the images have ground-truth. Again, due to intra-class variation, we avoid using color features for this dataset.

Note that we first perform k-means clustering using GIST descriptor [29] and the proposed saliency co-fusion is then applied to each cluster independently. This is to reduce the intra-class variation. Otherwise, a wide diversity might cause unnecessary difficulties in the co-fusion process. Empirically, we set the target cluster size to be 10, i.e., on average each cluster contains 10 images.

B. Performance Improvement by Co-fusion

The key point of our proposed saliency co-fusion process is to generate a fused saliency map that can better highlight the common object while suppressing the background saliency. To compare the quality of the fused saliency map with other saliency maps, we apply the simple segmentation approach, Otsu's method, on individual saliency maps of images in MSRC dataset, and report segmentation results in Table II. It can be seen that our method achieves about 10% gain over that of the best saliency extraction method [46]. Table II also shows the results of simple averaging or taking the maximum of those individual saliency maps at the pixel level also outperform the best single saliency map, clearly suggesting the advantage of using multiple saliency maps. Our method outperforms the simple average function and the max function by about 6% and 8%, respectively. Note that the Avg Jaccard Similarity value of 0.688 on MSRC dataset by using simple Otsu's method on the fused saliency map (without any parameter tuning) is even better than the result of 0.68 (see Table VII) obtained by [16] which used complex co-labeling, parameter tuning, and Grabcut.

Fig. 5 shows the visual comparison of individual saliency maps used and our fused saliency map. It can be seen that pixels pertaining to the woman (the common object) obtain boosted saliency values, while the background regions get suppressed saliency values in the final fused saliency maps which lead to clean segmentation results. Fig. 6 provides more examples of fused saliency maps and the corresponding segmentation results

TABLE XI
CLASS-WISE JACCARD SIMILARITY PERFORMANCE ON FLICKRMFC DATASET

	Apple picking	Baseball kids	Butterfly blossom	Cheetah safari	Cow pasture	Dog park	Dolphin aquarium	Fishing alaska	Gorilla zoo	Liberty statue	Parrot zoo	Stone henge	Swan zoo	Thinker robin
[13]	0.540	0.640	0.620	**0.850**	0.580	0.550	0.580	0.320	0.570	**0.900**	0.450	**0.960**	0.360	**0.840**
[12]	0.661	0.655	0.641	0.683	0.586	0.570	0.618	0.449	0.609	0.563	0.590	0.476	0.504	0.642
Ours	**0.720**	**0.783**	**0.729**	0.800	**0.694**	**0.700**	**0.717**	**0.663**	**0.631**	0.614	**0.640**	0.594	**0.604**	0.682

TABLE XII
CLASS-WISE JACCARD SIMILARITY PERFORMANCE ON COSEG-REP DATASET

	Repet -itive	Blue -flagris	Camel	Cormo -rant	Cranes -bill	Deer	Desert -rose	Dragon -fly	Egret	Fire -pink	Flea -bane	Forget -menot
[19]	0.754	0.890	0.641	0.493	0.842	0.450	0.880	0.380	0.463	**0.902**	**0.888**	**0.867**
[33]	0.747	0.823	0.688	0.592	0.854	0.634	0.826	**0.550**	0.499	0.781	0.829	0.842
Ours	**0.776**	**0.903**	**0.702**	**0.613**	**0.863**	**0.636**	**0.841**	0.542	**0.601**	0.884	0.851	0.849
	Frog	Geran-ium	Ostrich	Pear blossom	Piegon	Seagull	Seastar	Silen clorata	Snow owl	White campion	Wild beast	
[19]	0.484	0.897	0.605	0.777	0.427	0.464	0.631	**0.835**	0.355	0.739	0.839	
[33]	0.714	0.852	0.668	0.775	0.624	0.681	0.762	0.766	0.736	0.794	0.776	
Ours	**0.741**	**0.912**	**0.747**	**0.791**	**0.675**	**0.719**	**0.821**	0.828	**0.748**	**0.901**	**0.877**	

on MSRC and iCoseg datasets. Furthermore, it also shows the difference maps against the ground-truths.

C. Discussion on the Parameters

In Table III, we report our results obtained by fixing the parameter τ in (16) to 0.75 on all the datasets with GrabCut segmentation, and also the results obtained using simple Otsu's method. Due to the fact that categories of Internet images dataset are quite large, their results on each category are separately shown. We can see that even the simple Otsu's method is able to produce decent results with our fused saliency maps. This can be attributed to the high-quality saliency maps produced by our saliency co-fusion approach. By using GrabCut for segmentation, the performance of our method can be further improved. For parameter λ in (1), we empirically set it to 9. Also, we empirically set parameter γ in (3)–(5), and (15) to 1/300, and parameter θ in (7) and (10) to 0.75. Parameter ϕ^n in (16) is automatically computed using Ostu's method.

In order to examine the sensitivity of our method on different superpixel extraction methods and different parameter settings, we further conducted experiments using irregular superpixes generated by [49]. The results on MSRC dataset show that use of the superpixels of [49] with the global thresholding achieves the average Jaccard Similarity of 0.6876. However, this is almost same as the result of 0.6875 obtained by using SLIC [34]. We also vary the region-size parameter of SLIC [34]. By varying the region-size parameter of SLIC [34] from 20 to 100, the results can be seen in Table IV. It can be seen that the performance decreases only slightly with the increase of the region size. Therefore, these experiments indicate that the proposed method is robust to different super-pixel methods/settings.

D. Experiments for Comparison

For different datasets, we compare our method with the methods that report the state-of-the-art performance on the datasets. We denote "Ours (default)" as our method with the setting $\tau = 0.75$ using GrabCut while denoting "Ours (tuned)" as the one where we tune parameter τ with a step size of 0.03 from 0.60 to 0.99 over each category and report the best results, which is similar to other methods. Our method outperforms the state-of-the-art methods on two of the single object co-segmentation datasets (Coseg-Rep and Internet images) as shown in Tables V and VI. Also, some sample visual results of our method on Coseg-Rep dataset and Internet images dataset are shown in Figs. 7 and 8, respectively.

Note that for the Internet images dataset, since each of its categories consists of large number of images, we tune parameter τ per cluster. It can be seen from Tables V and VI that, in terms of Jaccard Similarity metric, our method achieves about 5% on Coseg-Rep dataset, 13%, 11%, and 9% improvements on Car, Horse and Airplane categories of Internet images dataset, respectively, when compared with the best results reported in [33] and [16] for CosegRep and Internet Images datasets, respectively. Table VII compares the results of our method with those of state-of-the-art methods on MSRC and iCoseg datasets. It can be seen that our results are competitive to the best one by [20], while our method is much faster than [20]. Specifically, running on the same PC with Intel Core i5-3470@3.20 GHz CPU and 32 GB RAM, [20] (using their own source codes in Matlab) takes 29.2 h to complete the entire segmentation process on MSRC dataset. However, our method (also in Matlab codes) takes only 8.5 h. These durations include the time taken for preprocessing steps as well like generating proposals in [20] and generating saliency maps in our method.

It is interesting to see that our method can also well handle Flickr MFC dataset that contains multiple common objects across the images and the repetitive category of Coseg-Rep dataset that contains repeated instances of objects, as shown in Tables VIII and XII, respectively. Our method with tuning per category outperformed the best one [12] (with supervised information) by 6% in terms of Jaccard Similarity metric despite

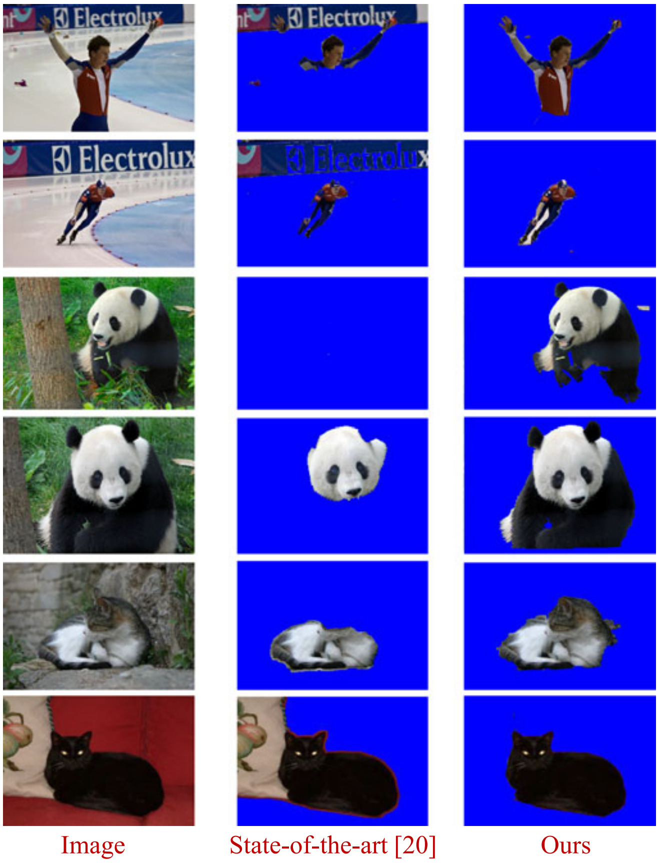

Fig. 10. Sample segmentation results where our method outperforms [20].

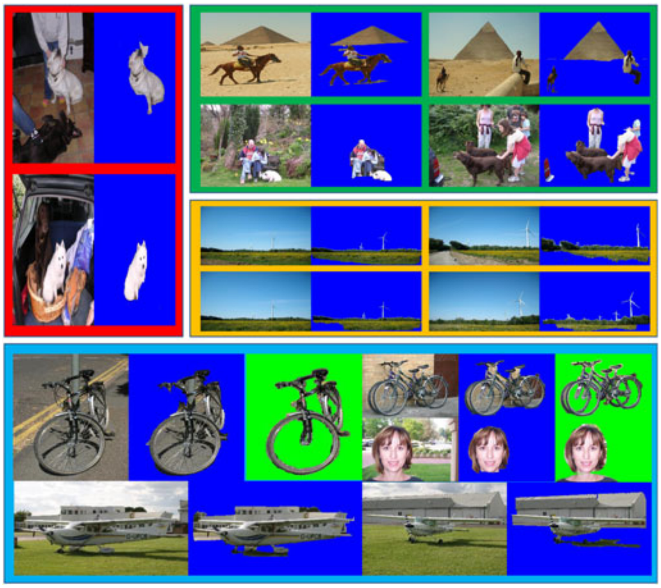

Fig. 11. Failure cases: our method fails (i) (red box) when a common object (black dog) is not salient in any of the saliency maps; (ii) (green box) when multiple foregrounds are present and the goal is to extract a particular foreground; and (iii) (yellow box) when very similar images are grouped for the co-segmentation process. (iv) (blue box): examples to show the limitations of our method in some specific categories in MSRC where our methods tend to segment convex shapes instead of thin rods in the bike class, miss segmenting the unsalient shoulder in the face class, and include the airport in the airplane class. Note that segmentations with a blue background are our results and those with a green background are the ground-truth results.

being an unsupervised method. In fact, our method's default setting itself outperforms the state-of-the-art method on Flickr MFC dataset. It should be noted that the comparison here is in terms of foreground/background segregation, and not multi-label segmentation. Fig. 9 shows some sample segmentation results in such multiple-foreground scenario. It can be seen that although different multiple objects are present in one category of the dataset, our method successfully extracts the foreground. As far as the repetitive scenario is concerned, our method obtained a Jaccard Similarity value of 0.776 in comparison to 0.754 obtained by [19] on the repetitive category of the Coseg-Rep dataset (see bottom three rows of Fig. 7 for such sample visual results).

Tables IX–XII list out the detailed Jaccard Similarity results of our method as well as the state-of-the-art methods on individual categories of the four datasets. It was seen earlier that our method performs worse than [20] on MSRC and iCoseg datasets as far as the overall average performance is concerned. The main reason could be that our method relies on saliency co-fusion. If the common object cannot be identified as salient by any of the saliency extraction methods, our method would not be able to segment it out. Interestingly, these tables reveal that despite such slightly inferior overall performance, our method outperforms [20] in 4 out of 14 and 15 out of 30 categories in MSRC and iCoseg datasets, respectively. Fig. 10 gives some visual examples of those categories, where our results look better than those of [20].

E. Limitations and Discussions

Although our method performs well on the benchmark datasets in general, there are some failure cases: (i) As shown in the red-box of Fig. 11, our method only segments out one dog and misses the other. This is because one of the dogs is extremely salient in all the saliency maps, while the other dog is not very salient in any of the saliency maps. (ii) Another case is when there are multiple common salient objects in the images, while the goal of benchmark dataset is to segment out only one common object. For such case, our method will segment out all the salient common objects as shown in the green-box of Fig. 11. (iii) Similar to almost all the co-segmentation methods, our method requires sufficient background variations across the images in one cluster. If very similar images are being included in one cluster, our method will fail to distinguish background from the foreground, as illustrated in the yellow-box of Fig. 11.

The blue box in Fig. 11 gives some class-specific examples where our method does not perform well. For example, (a) Bicycles in the bike category need segmentation of thin rods and tires whereas our method segments such bicycles into convex shapes such as triangles and disks due to using GrabCut; (b) Our method misses segmenting out shoulders in most of the images in the face category, because shoulders are not so salient; and (c) Many images in the plane category also include airports along with the planes, thus making it difficult to segment out the planes clearly.

V. Conclusion

We have proposed a novel saliency co-fusion approach for the purpose of image co-segmentation which uses the association of similar images to fuse multiple saliency maps of an image in order to boost up common foreground saliency and suppress background saliency. Experimental results on five benchmark datasets show that our method while co-fusing eight different saliency maps, achieves very competitive performance, compared to the state-of-the-art methods of image co-segmentation.

Acknowledgment

This research was carried out at the Rapid-Rich Object Search (ROSE) Laboratory, Nanyang Technological University, Singapore. The ROSE Laboratory is supported by the National Research Foundation, Prime Minister's Office, Singapore, under its IDM Futures Funding Initiative and administered by the Interactive and Digital Media Programe Office.

References

[1] C. Rother, T. Minka, A. Blake, and V. Kolmogorov, "Cosegmentation of image pairs by histogram matching-incorporating a global constraint into MRF," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2006, vol. 1, pp. 993–1000.

[2] D. S. Hochbaum and V. Singh, "An efficient algorithm for cosegmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep.-Oct. 2009, pp. 269–276.

[3] L. Mukherjee, V. Singh, and C. R. Dyer, "Half-integrality based algorithms for cosegmentation of images," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2009, pp. 2028–2035.

[4] A. Joulin, F. Bach, and J. Ponce, "Discriminative clustering for image co-segmentation," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2010, pp. 1943–1950.

[5] G. Kim, E. P. Xing, L. Fei-Fei, and T. Kanade, "Distributed cosegmentation via submodular optimization on anisotropic diffusion," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 169–176.

[6] H. Li, F. Meng, and K. N. Ngan, "Co-salient object detection from multiple images," *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 1896–1909, Dec. 2013.

[7] F. Meng, H. Li, G. Liu, and K. N. Ngan, "Object co-segmentation based on shortest path algorithm and saliency model," *IEEE Trans. Multimedia*, vol. 14, no. 5, pp. 1429–1441, Oct. 2012.

[8] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, "iCoseg: Interactive co-segmentation with intelligent scribble guidance," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2010, pp. 3169–3176.

[9] M. D. Collins, J. Xu, L. Grady, and V. Singh, "Random walks based multi-image segmentation: Quasiconvexity results and GPU-based solutions," in *Proc., IEEE Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 1656–1663.

[10] D. Batra, D. Parikh, A. Kowdle, T. Chen, and J. Luo, "Seed image selection in interactive cosegmentation," in *Proc. IEEE Int. Conf. Image Process.*, Nov. 2009, pp. 2393–2396.

[11] F. Meng, B. Luo, and C. Huang, "Object co-segmentation based on directed graph clustering," in *Proc. IEEE Visual Commun. Image Process.,* Nov. 2013, pp. 1–5.

[12] Z. Liu, J. Zhu, J. Bu, and C. Chen, "Object cosegmentation by nonrigid mapping," *Neurocomputing*, vol. 135, pp. 107–116, 2014.

[13] T. Ma and L. J. Latecki, "Graph transduction learning with connectivity constraints with application to multiple foreground cosegmentation," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1955–1962.

[14] G. Kim and E. P. Xing, "On multiple foreground cosegmentation," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 837–844.

[15] F. Meng, J. Cai, and H. Li, "On multiple image group cosegmentation," in *Proc. Asian Conf. Comput. Vis.*, 2014, pp. 258–272.

[16] M. Rubinstein, A. Joulin, J. Kopf, and C. Liu, "Unsupervised joint object discovery and segmentation in internet images," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1939–1946.

[17] M. Guillaumin, D. Kttel, and V. Ferrari, "Imagenet auto-annotation with segmentation propagation," *Int. J. Comput. Vis.*, vol. 110, no. 3, pp. 328–348, 2014.

[18] K. R. Jerripothula, J. Cai, and J. Yuan, "Group saliency propagation for large scale and quick image co-segmentation," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2015, pp. 4639–4643.

[19] J. Dai, Y. N. Wu, J. Zhou, and S.-C. Zhu, "Cosegmentation and cosketch by unsupervised learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1305–1312.

[20] A. Faktor and M. Irani, "Co-segmentation by composition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1297–1304.

[21] S. Vicente, C. Rother, and V. Kolmogorov, "Object cosegmentation," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 2217–2224.

[22] S. Vicente, V. Kolmogorov, and C. Rother, "Cosegmentation revisited: Models and optimization," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 465–479.

[23] J. C. Rubio, J. Serrat, A. López, and N. Paragios, "Unsupervised co-segmentation through region matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 749–756.

[24] L. Mukherjee, V. Singh, and J. Peng, "Scale invariant cosegmentation for image groups," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 1881–1888.

[25] J. Yuan *et al.*, "Discovering thematic objects in image collections and videos," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2207–2219, Apr. 2012.

[26] F. Wang, Q. Huang, and L. J. Guibas, "Image co-segmentation via consistent functional maps," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 849–856.

[27] A. Ion, J. Carreira, and C. Sminchisescu, "Image segmentation by figure-ground composition into maximal cliques," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2110–2117.

[28] D. E. Jacobs, D. B. Goldman, and E. Shechtman, "Cosaliency: Where people look when comparing images," in *Proc. ACM Symp. User Interface Softw. Technol.*, 2010, pp. 219–228.

[29] H.-T. Chen, "Preattentive co-saliency detection," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 1117–1120.

[30] H. Li and K. N. Ngan, "A co-saliency model of image pairs," *IEEE Trans. Image Process.*, vol. 20, no. 12, pp. 3365–3375, Dec. 2011.

[31] K.-Y. Chang, T.-L. Liu, and S.-H. Lai, "From co-saliency to co-segmentation: An efficient and fully unsupervised energy minimization model," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 2129–2136.

[32] H. Fu, X. Cao, and Z. Tu, "Cluster-based co-saliency detection," *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3766–3778, Oct. 2013.

[33] K. R. Jerripothula, J. Cai, F. Meng, and J. Yuan, "Automatic image co-segmentation using geometric mean saliency," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2014, pp. 3282–3286.

[34] R. Achanta *et al.*, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[35] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2014, pp. 280–287.

[36] K. Tang, A. Joulin, L.-J. Li, and L. Fei-Fei, "Co-localization in real-world images," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2014, pp. 1464–1471.

[37] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309-314, Aug. 2004.

[38] M.-M. Cheng, V. A. Prisacariu, S. Zheng, P. H. Torr, and C. Rother, "DenseCut: Densely connected CRFS for realtime grabcut," *Comput. Graph. Forum*, vol. 34, no. 7, pp. 193-201, 2015.

[39] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1155–1162.

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

[41] 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*, Dec. 2013, pp. 2976–2983.

[42] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 853–860.

[43] R. Margolin, A. Tal, and L. Zelnik-Manor, "What makes a patch distinct?" in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1139–1146.

[44] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2189–2202, Nov. 2012.

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

[46] H. Jiang *et al.*, "Salient object detection: A discriminative regional feature integration approach," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 2083–2090.

[47] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 1–15.

[48] P. Jaccard, "The distribution of the flora in the alpine zone," *New Phytologist*, vol. 9, pp. 37–50, 1912.

[49] A. Vedaldi and S. Soatto, "Quick shift and kernel methods for mode seeking," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 705–718.

[50] A. Joulin, F. Bach, and J. Ponce, "Multi-class cosegmentation," in *Proc. IEEE Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 542–549.

Koteswar Rao Jerripothula (S'15) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Roorkee, India, in 2012, and is currently working toward the Ph.D. degree at the Nanyang Technological University, Singapore.

His research interests include computer vision and multiumedia applications.

Jianfei Cai (S'97–M'02–SM'07) received the Ph.D. degree from the University of Missouri-Columbia, Columbia, MO, USA, in 2002.

He is currently an Associate Professor and has served as the Head of the Visual and Interactive Computing Division and the Head of the Computer Communication Division, School of Computer Engineering, Nanyang Technological University, Singapore. He has authored or coauthored more than 170 technical papers in international journals and conferences. His current research interests include computer vision, visual computing, and multimedia networking.

Prof. Cai has been actively participating in program committees of various conferences. He has served as the leading Technical Program Chair for the IEEE International Conference on Multimedia & Expo 2012 and the leading General Chair for the Pacific-Rim Conference on Multimedia 2012. Since 2013, he has been an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING. He has also served as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 2006 to 2013.

Junsong Yuan (S'06 M'08–SM'14) received the Graduate degree from the Special Class for the Gifted Young, Huazhong University of Science and Technology, Wuhan, Hubei, China, in 2002, the M.Eng. degree from the National University of Singapore, Singapore, in 2005, and the Ph.D. degree from Northwestern University, Evanston, IL, USA, in 2009.

He is an Associate Professor and Program Director of Video Analytics with the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore. He has authored or coauthored 3 books, 5 book chapters, and 150 conference and journal papers, and filed several patents, with technology licensed by the industry. His research interests include computer vision, video analytics, gesture and action analysis, large-scale visual search, and mining.

Prof. Yuan is a Program Co-Chair of the IEEE Conference on Visual Communications and Image Processing, Organizing Co-Chair of the Asian Conference on Computer Vision (ACCV'14), the Area Chair of ACCV'14, WACV'14, and ICME'14 and '15, and the Co-Chair of six workshops at CVPR/ICCV/SIGGRAPH Asia. He serves as a Guest Editor of the *International Journal of Computer Vision,* and is currently an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and *The Visual Computer Journal*. He was the recipient of the Nanyang Assistant Professorship and the Tan Chin Tuan Exchange Fellowship from the Nanyang Technological University, the Outstanding EECS Ph.D. Thesis Award from Northwestern University, the Doctoral Spotlight Award from the IEEE CVPR'09, and the National Outstanding Student Award from the Ministry of Education, China.