Interactive Cosegmentation Using Global and Local Energy Optimization

Xingping Dong, Jianbing Shen, *Senior Member, IEEE*, Ling Shao, *Senior Member, IEEE*, and Ming-Hsuan Yang, *Senior Member, IEEE*

***Abstract*— We propose a novel interactive cosegmentation method using global and local energy optimization. The global energy includes two terms: 1) the global scribbled energy and 2) the interimage energy. The first one utilizes the user scribbles to build the Gaussian mixture model and improve the cosegmentation performance. The second one is a global constraint, which attempts to match the histograms of common objects. To minimize the local energy, we apply the spline regression to learn the smoothness in a local neighborhood. This energy optimization can be converted into a constrained quadratic programming problem. To reduce the computational complexity, we propose an iterative optimization algorithm to decompose this optimization problem into several subproblems. The experimental results show that our method outperforms the state-of-the-art unsupervised cosegmentation and interactive cosegmentation methods on the iCoseg and MSRC benchmark data sets.**

***Index Terms*— Co-segmentation, Gaussian mixture model, optimization, local spline regression, histogram matching.**

I. Introduction

With the development of mobile cameras, users can easily capture more and more images and share them on the Internet. Among a group of images, the same or similar foreground objects are likely to occur. The goal of image co-segmentation is to exploit information from multiple images to identify the foreground objects with pixel-wise accuracy. Rother *et al.* [12] proposed an image co-segmentation method using the histogram matching of the foreground within the Markov Random Field (MRF) framework for segmentation. Early co-segmentation approaches [12], [17], [18], [20] only used a pair of images as input under the assumption of sharing a common foreground object. Numerous approaches [19], [25], [26], [28], [30], [31], [35] have recently been developed to co-segment multiple images. All these unsupervised co-segmentation approaches have achieved more accurate results than the classic single-image segmentation methods. However, these unsupervised co-segmentation methods do not perform well when the foreground and background are similar in one image, or when the backgrounds among images are similar as it is difficult to find the common object automatically.

Scribbles for foreground and background pixels have been used to improve the image co-segmentation performance. Batra *et al.* [21] added user scribbles in some of input images to build two global Gaussian Mixture Models (GMM) for both foreground and background classes. A graph cut algorithm is then used to co-segment these images. However, these GMMS models are less effective where the foreground and background are similar. For these images, it is necessary to add more scribbles that provide supervised information to indicate the foreground objects in these similar regions. In contrast to the co-segmentation approaches within the MRF framework, Collins *et al.* [29] proposed an image co-segmentation method using the random walker algorithm where the smooth term is based on normalized Euclidean distance of pixel intensity. However, this method is sensitive to parameter settings and likely to generate different segmentation results [22].

To address the above-mentioned issues, we propose a novel interactive co-segmentation algorithm using the global and local energy optimization. Our energy function includes three terms including global scribbled energy, inter-image energy, and local smooth energy. The first two global energy terms are used to reduce the user scribbles including those newly added scribbles by GMM, and the last local smooth energy is used to solve the problem that the parameters are sensitive in the smooth term. Both foreground GMM and background GMM are first built by the user scribbles in all images, which can be viewed as the global guide information from users. Our global scribbled energy is constructed based on the superpixels with highest foreground/background posterior probability from each image. Assuming each image has a common foreground histogram in a group, we use an inter-image energy to compare them to an average histogram. By considering the consistence

Manuscript received May 6, 2015; accepted July 5, 2015. Date of publication July 14, 2015; date of current version July 31, 2015. This work was supported in part by the National Basic Research Program of China (973 Program) under Grant 2013CB328805, in part by the National Natural Science Foundation of China under Grant 61272359, in part by the Program for New Century Excellent Talents in University under Grant NCET-11-0789, and in part by the Fok Ying-Tong Education Foundation for Young Teachers. The work of M.-H. Yang was supported in part by the National Science Foundation CAREER under Grant 1149783, in part by NSF through the Division of Information and Intelligent Systems under Grant 1152576, and in part by Specialized Fund for Joint Building Program of Beijing Municipal Education Commission. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Kiyoharu Aizawa. *(Corresponding author: Jianbing Shen.)*

X. Dong and J. Shen are with the Beijing Laboratory of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, Beijing 100081, China (e-mail: dongxingping@bit.edu.cn; shenjianbing@bit.edu.cn).

L. Shao is with the Department of Computer Science and Digital Technologies, Northumbria University, Newcastle upon Tyne NE1 8ST, U.K. (e-mail: ling.shao@ieee.org).

M.-H. Yang is with the School of Engineering, University of California, Merced, CA 95344 USA (e-mail: mhyang@ucmerced.edu).

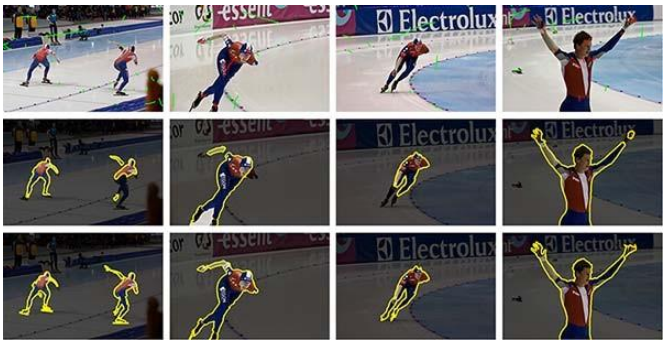


Fig. 1. Co-segmentation results. Top: input skate group images. Middle: results by the random walker co-segmentation [29]. Bottom: results by our approach. Note that all the co-segmentation results are produced by the same scribbles (red for foreground and green for background), where four representative images are shown from the total thirteen input images.

and the smoothness of superpixels in a neighborhood, the local energy is then designed as the local smooth term. The spline regression is further employed to automatically learn this local smooth term, which avoids tuning the parameters of the general Gaussian smooth function. The minimization problem of our energy function can be converted into the constrained quadratic programming (QP) problem, where an iterative optimization strategy is designed for the computational efficiency. As shown in Fig. 1, parts of the background and foreground objects are similar in color. Since there are many strong edges in the common object, the random walker cannot reach the right segmentation edges without enough user scribbles. In contrast, our approach performs well in these images as shown in the last row of Fig. 1, where our energy captures the local and global characteristics of the foreground objects after optimization. The source code of this work will be available at http://github.com/shenjianbing/interactivecoseg.

The contributions of this work to co-segmentation are summarized as follows:

1) We propose a novel energy optimization method for interactive co-segmentation including global scribbled energy, local smooth energy, and inter-image energy.
2) The spline regression is introduced to design the smooth term, which avoids tuning the parameters of the smooth term and has better self-adaptability to most complex natural images.
3) An iterative optimization algorithm using a constrained QP solver is presented for the computational efficiency which scales up well for large datasets.

II. Related Work

Most image co-segmentation methods are derived from single-image segmentation methods by adding similar foreground constraints in the MRF based optimization framework. Similar to the single-image segmentation, the current image co-segmentation approaches can be classified into two groups: unsupervised and interactive co-segmentation.

Unsupervised Co-Segmentation: Rother *et al.* [12] introduced an image co-segmentation method by combining the MRF framework and global constraints with foreground histogram matching. Based on a pair of images, the co-segmentation problem is posed as an energy minimization problem, and a graph cut method based on the trust region is proposed. Based on this work, Mu and Zhou [13] used the L_2 norm constraint as the global constraint instead of the L_1 norm in [12]. Hochbaum and Singh [18] proposed a max-flow algorithm by modifying the histogram matching. In addition to histogram-matching based algorithms, clustering has also been utilized for co-segmentation. Joulin *et al.* [19] combined normalized cuts and kernel methods to design a discriminative clustering co-segmentation framework, where they classified the pixels in all images into foreground and background classes. This co-segmentation method was solved by a continuous convex searching optimization. Recently, they extended their framework to multi-class co-segmentation [28]. The co-segmentation method for a large-scale image dataset was proposed in [24]. This method was modeled by temperature maximization with finite K heat sources on a linear anisotropic diffusion system. This can be formulated as a K-way segmentation that maximizes the segmentation confidence of every pixel in an image. In theory, this temperature function is a sub-modular function, and thus at least a constant approximation of the optimal solution is guaranteed by a greedy algorithm. These unsupervised co-segmentation methods do not perform well when the foreground objects and the background are similar. The interactive co-segmentation methods alleviate these problems by indicating the foreground objects with sparse scribbles.

Interactive Co-Segmentation: We first review the related work on interactive segmentation methods for a single-image. Boykov and Jolly [7] converted interactive segmentation into a discrete optimization problem, which was solved by graph cut. Sinop and Grady [14] proposed a seeded image segmentation framework by unifying graph cut and random walker. Within this framework, the graph cut or random walker [6], [34] is viewed as a certain energy minimization with an L_1 norm or an L_2 norm. Xiang *et al.* [22] proposed a semi-supervised classification algorithm via local spline regression, which can be used for the interactive image segmentation. Zhang and Ji [4] presented an interactive segmentation using the Bayesian network model, where they performed the superpixel over-segmentation to construct a multilayer Bayesian network. The idea of interactive segmentation using scribbles for a single-image can be naturally extended to interactive image co-segmentation. Batra *et al.* [21], [23] proposed an interactive co-segmentation technique, which enabled the user to correct the inconsistent segmentation by adding sparse scribbles. They proposed an recommendation method to help users choose the regions needing the scribbles. This algorithm assumes all images in a group share a common foreground GMM and a background GMM, which are represented by all scribbles. With the common GMMs, they process each image as a single-image segmentation [7]. Collins *et al.* [29] proposed an interactive co-segmentation approach by adding the consistency constraint between the foreground objects using the random walks model. However, the random walks

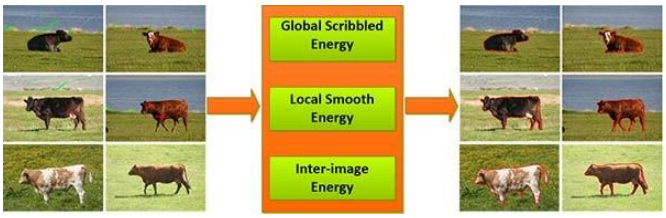
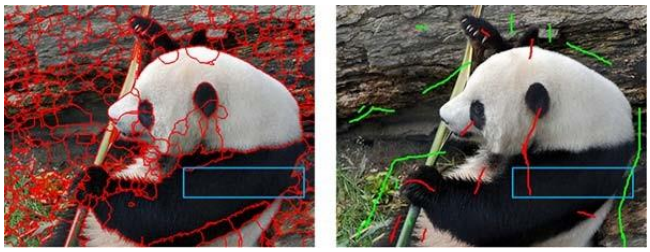


Fig. 2. Workflow of our algorithm. The left four images are user scribbled images where the red and green scribbles indicate the foreground and background respectively. There are no scribbles on the bottom two images since our algorithm can segment the objects of some images without any scribbles. The right six images are the results of co-segmenting all the 30 images in the cow class in the MSRC dataset.



(a) (b)

Fig. 3. Illustration of processing the conflicting superpixel between foreground scribbles and background scribbles. (a) The image of superpixels; (b) the user scribbled image.

optimization will make the co-segmentation results sensitive to the quantities and positions of the user scribbles [6].

III. Proposed Algorithm

The workflow of our interactive co-segmentation framework using global and local energy minimization is shown in Fig. 2. The user is just required to indicate the sparse scribbles on a small number of images which contain the common object. Then we denote this group of n images by $\{I_1, I_2, \ldots, I_l, I_{l+1}, \cdots, I_n\}$. The first l images are the scribbled images and the others are the unscribbled images. We pre-segment each image I_i into a group of small regions $R_i = \{r_{i,j}\}_{j=1,\ldots,m_i}$ (i.e. superpixels) by an over-segmentation method such as the mean-shift algorithm [8], where m_i is the number of superpixels in image I_i. Two kinds of features are extracted from these superpixels R_i. One is the average color intensities $X_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,m_i}] \in R^{d \times m_i}$, where each column $x_{i,j}$ of X_i is the mean color intensities of a superpixel $r_{i,j}$. The other is the color histogram descriptors $H_i = [h_{i,1}, h_{i,2}, \ldots, h_{i,m_i}] \in R^{d_h \times m_i}$ by color intensities.

A vector $y_i \in \{0,1\}^{m_i}$ is used to represent the superpixel labels for image I_i, where 1 or 0 assigns the corresponding superpixel to the foreground or the background. Then the foreground histogram of I_i is calculated as $h_i^f = \sum_j h_{i,j} y_i(j)$, i.e. $h_i^f = H_i \cdot y_i$.

For each scribbled image $I_i (i \le l)$, we use two index vectors $y_i^f \in \{0,1\}^{m_i}$ and $y_i^b \in \{0,1\}^{m_i}$ to indicate the scribbled foreground or background superpixels respectively. There will be a conflict when two kinds of scribbles fall into the same superpixel. This conflicting superpixel may occur at a weak boundary, which may consist of most foreground pixels and a few background pixels such as the superpixel in the blue rectangle of Fig. 3. Our goal is to co-segment the foreground, and the segmentation results should contain the foreground information as much as possible. If we assign this surperpixel to the background, some foreground information will be lost. Thus, the conflicting superpixel is assigned to the foreground. Then, these vectors can be formulated as follows: if foreground scribbles fall into superpixel $r_{i,j}$, $y_i^f(j) = 1$, else $y_i^f(j) = 0$; if background scribbles fall into superpixel $r_{i,j}$ and foreground scribbles do not fall into superpixel $r_{i,j}$, $y_i^b(j) = 1$, else $y_i^b(j) = 0$. The problem of interactive co-segmentation can be formulated as energy minimization. Our energy function consists of three components: the global scribbled energy, the local smooth energy, and the inter-image energy.

$$E = E_{smooth} + \lambda_1 E_{\mathrm{GMM}} + \lambda_2 E_{inter}, \tag{1}$$

where λ_1, λ_2 are the trade-off parameters.

A. Global Scribbled Energy

How to effectively utilize the user scribbles is key for interactive co-segmentation. We build a Gaussian Mixture Model as a global guide, where the GMM is generated from the scribbled regions of all images and then it is used as global guidance for co-segmentation. The general GMMs with five components [9] are used in our approach, and these GMMs are isotropic without specific covariance forms. We can assume that all images in the group share a common model, i.e., only one model need to be learned. It can be represented by using the global GMM which consists of a foreground GMM (denoted as GMM$_f$) and a background GMM (denoted as GMM$_b$). We denote the color intensities of all scribbled foreground and background superpixels by X_s^f and X_s^b. These two models are separately learned by X_s^f and X_s^b. More details concerning the learning process can be seen in [15].

Using the global GMM$_f$ and GMM$_b$, we can obtain the posterior probability of foreground $P_i^f \in [0,1]^{m_i}$ and background $P_i^b \in [0,1]^{m_i}$ in each image. The posterior probability may not be very accurate for the foreground or the background, since some superpixels in the foreground are similar to those in the background. Thus, we choose the superpixels with K highest posterior probabilities as guidance to reduce the error. We use two index vectors $y_i^{\mathrm{GMM}_f} \in \{0,1\}^{m_i}$ and $y_i^{\mathrm{GMM}_b} \in \{0,1\}^{m_i}$ to indicate K highest posterior probabilities of the foreground and the background, which store the indexes of these superpixels.

We define the global scribbled energy, which measures the consistence between the superpixels with K highest posterior probabilities and their corresponding labels, as follows:

$$E_{\mathrm{GMM}} = \sum_{i=1}^{n} \left(\sum_{y_i^{\mathrm{GMM}_f}(j)=1} \| y_i(j) - 1 \|^2 + \sum_{y_i^{\mathrm{GMM}_b}(j)=1} \| y_i(j) \|^2 \right). \tag{2}$$

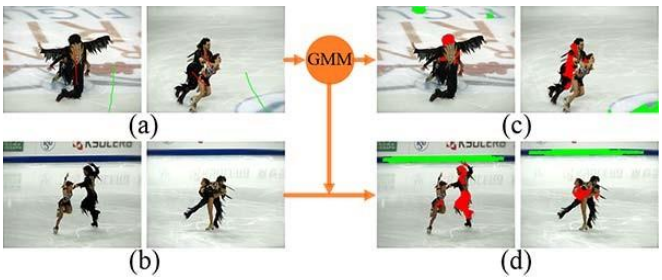


Fig. 4. Illustration of the labeled scribbles by GMM: (a) the user scribbled images. (b) The images without user scribbles. (c) and (d) The automatically labeled scribbles by GMM. In (c) and (d), the red and green regions indicate the foreground and background respectively.

This energy function is transformed to

$$E_{\mathrm{GMM}} = \sum_{i=1}^{n} (y_i - y_i^{\mathrm{GMM}})^{\mathsf{T}} D_i (y_i - y_i^{\mathrm{GMM}}), \tag{3}$$

where $y_i^{\mathrm{GMM}} = y_i^{\mathrm{GMM}_f} \cdot 1 + y_i^{\mathrm{GMM}_b} \cdot 0$, $D_i \in R^{m_i \times m_i}$ is a diagonal matrix, and the diagonal elements correspond to vector $y_i^{\mathrm{GMM}_f} + y_i^{\mathrm{GMM}_b}$, i.e. $D_i = diag(y_i^{\mathrm{GMM}_f} + y_i^{\mathrm{GMM}_b})$.

Note that y_i^{GMM} is the label vector given by GMM. In other words, the GMM can be viewed as a "user" who labels $2K$ superpixels including the foreground and background in each image automatically, and the labels are stored in vector y_i^{GMM}. Then D_i indicates the superpixels that are labeled by GMM. Fig. 4 shows the labeled scribbles by GMM. These guiding scribbles may be added in foreground (background) regions according to the most different appearance of the background (foreground). With the help of these guiding scribbles, the user can reduce scribbles in these regions. The user only needs to add scribbles in these regions for a few images. These regions of other images in the same group will be labeled automatically by GMMs.

B. Local Smooth Energy

The local smooth energy considers the local smoothness of labels, i.e., the labels will be the same in a local neighborhood, when their corresponding features are similar. Instead of the commonly-used Gaussian function with affinity measurement for smoothness in previous work [14], [29], we use the spline regression to learn the local smoothness. The spline consists of polynomials and Green functions. It is smooth, nonlinear, and is able to interpolate the scattered data points with high accuracy [22]. This smooth energy can be viewed as the local energy inside each image. Thus, we only need to consider the smoothness in a single image. For the precise description of notation, we redefine some notations. We denote c_i as the feature, z_i as the label of superpixel R_i in a single image, and m as the number of superpixels. The spatial adjacent neighborhood of each superpixel R_i (including itself) is denoted as $N_i = \{c_{i_j} \mid j = 1, 2, \ldots, k_i\}$, where subscript i_j stands for an index, i.e., $i_j \in 1, 2, \ldots, m$. Then, in each neighborhood we assume that there exists a spline function $g_i : R^d \to R$ which can directly map each pixel c_{i_j} to its label z_{i_j}, i.e $z_{i_j} = g_i(c_{i_j}), j = 1, 2, \ldots, k_i$. According to this assumption, we can build a general objective function for spline regression in its spatial adjacent neighborhood:

$$J(g_i) = \sum_{j=1}^{k_i} (z_{i_j} - g_i(c_{i_j}))^2 + \lambda S(g_i), \tag{4}$$

where $S(g_i)$ is a penalty function, λ is a trade-off parameter, and $J(g_i)$ denotes the loss energy of regularization.

For a small λ, this objective function can be transformed to a simplified formulation. The details can be seen in Appendix A. Here, we directly give this formulation:

$$J(g_i) \approx \lambda Z_i^{\mathsf{T}} M_i Z_i, \tag{5}$$

where M_i is the upper left $k_i \times k_i$ submatrix of the inverse matrix of the coefficient matrix in (24), and λ is a small value, Z_i is the label vector, i.e., $Z_i = [z_{i_1}, z_{i_2}, \ldots, z_{i_{k_i}}]^{\mathsf{T}}$.

After obtaining the loss energy in each neighborhood N_i, the loss energies can be added together as a global energy. We can ignore coefficient λ for computational simplicity in (5) as follows:

$$E(y) \propto \sum_{i=1}^{m} Z_i^{\mathsf{T}} M_i Z_i = z^{\mathsf{T}} M z, \tag{6}$$

where $z = [z_1, z_2, \ldots, z_m]^{\mathsf{T}} \in R^m$, $S_i \in R^{k_i \times m}$ is a row selection matrix subjecting to $Z_i = S_i z$, and $M = \sum_{i=1}^{m} S_i^{\mathsf{T}} M_i S_i$.

For one scribbled image, we add the scribbled information to (6) as a constraint and transform it to the following minimization:

$$\begin{aligned} &\min_z z^{\mathsf{T}} M z, \\ &s.t. \quad \begin{cases} z(j) = 1, & \text{if } c_j \text{ is scribbled as foreground} \\ z(j) = 0, & \text{if } c_j \text{ is scribbled as background.} \end{cases} \end{aligned} \tag{7}$$

Combining the aforementioned notations and (7), we obtain the local smooth energy of all images as follows:

$$\begin{aligned} &\min_{y_i} E_{smooth} = \sum_{i=1}^{n} y_i^{\mathsf{T}} M_i^{smooth} y_i, \\ &s.t. \quad \begin{cases} y_i(j) = 1, & \text{if } x_{i,j} \in X_s^f, \quad i = 1, 2, \ldots, l \\ y_i(j) = 0, & \text{if } x_{i,j} \in X_s^b, \quad i = 1, 2, \ldots, l \end{cases} \end{aligned} \tag{8}$$

where X_s^f or X_s^b is the set of all scribbled foreground or background, and M_i^{smooth} denotes M in (7).

C. Inter-Image Energy

The inter-image energy measures foreground similarity between different images in a group. In previous work [12], [17], [18], [20], [26], [27], [32], the histogram matching descriptor has been used to build the foreground model. The foreground similarity between different images can be measured by the distance of corresponding histograms. It is intuitive to compare the histograms between image regions, but the computation is expensive. We compare them to a common global foreground histogram $\bar{h}$ to reduce the computational complexity. The corresponding inter-image energy is formulated as follows:

$$E_{inter} = \sum_{i=1}^{n} \| h_i^f - \bar{h} \|^2 = \sum_{i=1}^{n} \| H_i y_i - \bar{h} \|^2, \tag{9}$$

where "$\cdot$" is the Euclidean distance, $H_i \in R^{d_h \times m_i}$, and $h_i^f \in R^{d_h}$ is the foreground histogram.

Given the histogram h_i^f of each foreground, we can achieve the optimum of $\bar{h}$, which is the center of h^f by setting the derivative of (9) to be zero. The formulation is defined as:

$$\bar{h} = \frac{1}{n} \sum_{i=1}^{n} h_i^f. \tag{10}$$

D. Total Energy Minimization

Reformulating the above scribbled energy and the inter-image energy, the total energy minimization can be converted into a quadratic programming problem. The scribbled energy in (3) can be reformulated as:

$$E_{\mathrm{GMM}} = \sum_{i=1}^{n} y_i^{\mathsf{T}} D_i y_i - 2 y_i^{\mathsf{T}} D_i y_i^{\mathrm{GMM}} + y_i^{\mathrm{GMM}\mathsf{T}} y_i^{\mathrm{GMM}}. \tag{11}$$

The inter-image energy in (9) can be reformulated as:

$$E_{inter} = \sum_{i=1}^{n} y_i^{\mathsf{T}} H_i^{\mathsf{T}} H_i y_i - 2 y_i^{\mathsf{T}} H_i^{\mathsf{T}} \bar{h} + \bar{h}^{\mathsf{T}} \bar{h}. \tag{12}$$

Furthermore, we can use bounds to limit y_i to the unit box as well as enforce the scribbles' constraints. Then, substituting (8), (11), and (12) into (1), we have:

$$\begin{aligned} \min_{y_i, \bar{h}} E = & \sum_{i=1}^{n} y_i^{\mathsf{T}} L_i y_i + \lambda_2 \bar{h}^{\mathsf{T}} \bar{h} + C_i \\ & - 2\lambda_2 y_i^{\mathsf{T}} H_i \bar{h} - 2\lambda_1 y_i^{\mathsf{T}} D_i y_i^{\mathrm{GMM}}, \\ s.t.\ & l_i \le y_i \le u_i, \quad i = 1, 2, \ldots, n, \end{aligned} \tag{13}$$

where $L_i = M_i^{smooth} + \lambda_1 D_i + \lambda_2 H_i^{\mathsf{T}} H_i$ and $C_i = \lambda_1 y_i^{\mathrm{GMM}\mathsf{T}} y_i^{\mathrm{GMM}}$. The 2-tuple (l_i, u_i) is $(1, 1)$ for foreground scribbles, $(0, 0)$ for background scribbles, and $(0, 1)$ otherwise. We can denote a long vector $Y = [y_1^{\mathsf{T}}, y_2^{\mathsf{T}}, \ldots, y_n^{\mathsf{T}}, \bar{h}^{\mathsf{T}}]^{\mathsf{T}} \in R^{N+d_h}$, $N = \sum_{i=1}^{n} m_i$.

The energy function (13) can be formulated as the following bound-constrained QP problem (omitting the constant C_i):

$$\begin{aligned} \min_{\tilde{Y}} & E = \tilde{Y}^{\mathsf{T}} \tilde{M} \tilde{Y} - 2\tilde{Y}^{\mathsf{T}} \tilde{V}, \\ s.t.\ & \tilde{l} \le \tilde{Y} \le \tilde{u} \end{aligned} \tag{14}$$

where $\tilde{V} = [v_1^{\mathsf{T}}, v_2^{\mathsf{T}}, \ldots, v_n^{\mathsf{T}}, 0]^{\mathsf{T}} \in R^{N+d_h}$, $v_i = \lambda_1 D_i y_i^{\mathrm{GMM}}$, $i = 1, 2, \ldots, n$, $\tilde{l} = [l_1^{\mathsf{T}}, l_2^{\mathsf{T}}, \ldots, l_n^{\mathsf{T}}, -\infty]^{\mathsf{T}} \in R^{N+d_h}$, $\tilde{u} = [u_1^{\mathsf{T}}, u_2^{\mathsf{T}}, \ldots, u_n^{\mathsf{T}}, +\infty]^{\mathsf{T}} \in R^{N+d_h}$, and

$$\tilde{M} = \begin{bmatrix} L_1 + \lambda_2 H_1^{\mathsf{T}} H_1 & & & -\lambda_2 H_1^{\mathsf{T}} \\ & \ddots & & \vdots \\ & & L_n + \lambda_2 H_n^{\mathsf{T}} H_n & -\lambda_2 H_n^{\mathsf{T}} \\ -\lambda_2 H_1 & \cdots & -\lambda_2 H_n & \lambda_2 n I \end{bmatrix}. \tag{15}$$

When the number of all superpixels is small, the above QP problem (14) can be solved by using some constrained QP optimization algorithms, such as interior-point [10] and active-set [5] algorithm. However, the computation of these algorithms will be expensive for co-segmenting large-scale groups of images. Thus, we propose an iterative optimization algorithm for efficient computation in the next sub-section.

E. Iterative Optimization Algorithm

We observe two properties of the proposed algorithm.

Property 1: Given histogram $\bar{h}$, the energy minimization (13) can be decomposed as some sub-problems:

$$\begin{aligned} \min_{y_i} & E_i = y_i^{\mathsf{T}} L_i y_i + y_i^{\mathsf{T}} \dot{V}_i + \dot{C}_i, \\ s.t.\ & l_i \le y_i \le u_i \quad i = 1, 2, \ldots, n, \end{aligned} \tag{16}$$

where $\dot{V}_i = -2\lambda_2 H_i^{\mathsf{T}} \bar{h} - 2\lambda_1 D_i y_i^{\mathrm{GMM}}$ and $\dot{C}_i = \lambda_2 \bar{h}^{\mathsf{T}} \bar{h} + C_i$. Each sub-problem is a constrained QP problem with a small scale. Then it can be solved with cheap computation, and we choose the interior-point algorithm to solve these sub-problems.

Property 2: Given vector $y_i, i = 1, 2, \ldots, n$, $\bar{h}$ can be easily calculated as follows:

$$\bar{h} = \frac{1}{n} \sum_{i=1}^{n} h_i^f = \frac{1}{n} \sum_{i=1}^{n} H_i y_i. \tag{17}$$

According to property 2, we only need to initialize y_i as y_i^0, then we can get the initialization of $\bar{h}$: $\bar{h}^0 = \frac{1}{n} \sum_{i=1}^{n} H_i y_i^0$.

According to the above two properties and the initialization, our iterative optimization algorithm is designed as follows. In each iteration, we first fix $\bar{h}$ and solve the ith sub-problem (16) to achieve y_i, then update $\bar{h}$ using (17) and repeat this process until $i = n$. This iteration is repeated until the final convergence.

The remainder problem is how to initialize y_i. According to (13), if we set $\lambda_2 = 0$ (i.e., omit the inter-image energy), the minimization problem is still feasible, since the local smooth energy keeps the local information and the scribbled energy keeps the global information. This solution is enough for initialization. This minimization problem is formulated as:

$$\begin{aligned} \min_{y_i} E = & \sum_{i=1}^{n} y_i^{\mathsf{T}} (M_i^{smooth} + \lambda_1 D_i) y_i - 2\lambda_1 y_i^{\mathsf{T}} D_i y_i^{\mathrm{GMM}} \\ s.t.\ & l_i \le y_i \le u_i, \quad i = 1, 2, \ldots, n. \end{aligned} \tag{18}$$

This problem can also be decomposed into some sub-problems similar to (16), and we can further relax the constraint to make the sub-problems unconstrained QP problems. For a scribbled image, we have

$$\begin{aligned} \min_{y_i} E_i = & y_i^{\mathsf{T}} (M_i^{smooth} + \lambda_1 D_i) y_i - 2\lambda_1 y_i^{\mathsf{T}} D_i y_i^{\mathrm{GMM}} \\ & + \lambda_0 (y_i - y_i^s)^{\mathsf{T}} D_i^s (y_i - y_i^s) \\ = & y_i^{\mathsf{T}} L_i^r y_i - 2 y_i^{\mathsf{T}} V_i^r + C_i^r \end{aligned} \tag{19}$$

where the scribble index matrix is $D_i^s = diag(y_i^f + y_i^b)$ (representing). $y_i^s = y_i^f \cdot 1 + y_i^b \cdot 0$, $L_i^r = M_i^{smooth} + \lambda_1 D_i + \lambda_0 D_i^s$, $V_i^r = \lambda_0 D_i^s y_i^s$, and $C_i^r = \lambda_0 y_i^{s\mathsf{T}} y_i^s$, for $i = 1, 2, \ldots, l$. The parameter λ_0 should be large enough to keep the accuracy of user scribbles.

Algorithm 1 Co-Segmentation by Hybrid Optimization

Input: A group of images I and some user scribbles U_n;
Output: The label vectors y_i, and the co-segmentation results $I^{(i)}$, $i = 1, 2, \cdots, n$;

Obtain the over-segmentation regions R_i of I_i;
Extract the color intensities X_i and histograms H_i;
Obtain the index vectors $y_i^f \in \{0,1\}^{m_i}$ and $y_i^b \in \{0,1\}^{m_i}$ from the scribbled image $I_i (i \leq l)$;
Generate the GMM from all scribbled superpixels and then get the label vector y_i^{GMM} and the index matrix D_i;
Initialize y_i using (21) and (22);
Initialize $E^{(2)}$ by substituting y_i and $\overline{h}$ into (13);
Repeat: $E^{(1)} = E^{(2)}$, $E^{(2)} = 0$, $t = t + 1$.
for i=1:n **do**
    Solve (16) using interior-point algorithm and obtain the solution as y_i^{new};
    Update $\overline{h} = \overline{h} + H_i(y_i^{new} - y_i)/n$, $y_i = y_i^{new}$;
    Calculate E using (16)
    Set $E^{(2)} = E^{(2)} + E$
end for
until: $\|E^{(1)} - E^{(2)}\| < 10^{-3}$ or $t > T_{max}$
Obtain the co-segmentation results of $\{I^{(1)}, ..., I^{(n)}\}$;

For an unscribbled image, the minimization can be formulated as:

$$\min_{y_i} E_i = y_i^{\mathsf{T}}(M_i^{smooth} + \lambda_1 D_i) y_i - 2\lambda_1 y_i^{\mathsf{T}} D_i y_i^{\mathrm{GMM}} = y_i^{\mathsf{T}} L_i^{r} y_i - 2\lambda_1 y_i^{\mathsf{T}} D_i y_i^{\mathrm{GMM}}, \quad (20)$$

where $L_i^{r} = M_i^{smooth} + \lambda_1 D_i$, for $i = l+1, l+2, \ldots, n$.

Based on (19) and (20), we can get the optimal solution of y_i as follows:

$$y_i = L_i^{r-1} V_i, \quad i = 1, 2, \ldots, l, \quad (21)$$

$$y_i = L_i^{r-1} D_i y_i^{\mathrm{GMM}}, \quad i = l+1, l+2, \ldots, n. \quad (22)$$

The threshold $t = 0.5$ is used to get the binary y_i, i.e., $y_i(j) = 1$, if $y_i(j) \geq 0.5$; $y_i(j) = 0$, otherwise. We note L_i^{r} in (21) or (22) is invertible for $i = 1, 2, \ldots, n$. This proof is shown in Appendix B. It is worth mentioning that our iterative optimization is guaranteed to converge. The theoretical proof is given in Appendix C. In our experiments, this iterative optimization usually converges after two or three iterations. Finally, we summarize the whole pseudo-code of the proposed algorithm in Algorithm 1.

IV. Experimental Results

In this section, we evaluate the proposed co-segmentation method on many image groups. The qualitative and quantitative comparison results between the state-of-the-art algorithms and ours are given in the following sub-sections. We used two benchmark datasets: MSRC dataset [11] and iCoseg dataset [21], which have been widely used by previous work to evaluate the performance of image co-segmentation methods. Both datasets include the ground truth segmentation masks that are used for the quantitative evaluation. Then we randomly select 10 object classes with about 30 images per class from MSRC, and 20 classes with a varying number of images per class from iCoseg in our experiments. We provided the new ground-truth images of 10 classes from MSRC, since the original ground-truth images are not accurate enough for our experiments. Furthermore, we used the large-scale image groups [36] for evaluating the computational efficiency of our approach, which include car (4347 images), horse (6381 images), and airplane (4542 images). 200 images with ground-truth from each group are randomly selected for experiments. In our experiments, the quantitative evaluation includes two performance metrics: precision P and Jaccard similarity J. P denotes the ratio of correctly labeled pixels for both foreground and background, and J represents the intersection over union of the segmentation results and ground truth masks. These two metrics are commonly used for performance evaluations of image co-segmentation approaches.

A. Parameter Settings

There are some suggestions for our interactive method on how to add user scribbles. Some images with a complex background should be added scribbles first, since these images can provide more information to build GMMs and it is unnecessary to add scribbles to images with a simple background such as the bottom images in Fig. 2. The scribble should contain as many colors as possible, so the regions with variable colors inside the foreground/background are good choices to add scribbles. The regions with similar colors between the foreground and the background should be scribbled. The user should add foreground (background) scribbles until these scribbles have contained most color information.

After adding scribbles, we setup the parameters of the mean-shift pre-segmentation algorithm [8] including the spatial bandwidth $h_s = 10$, the range bandwidth $h_r = 7$, and the minimum size of final output regions $M_{ms} = 30$. With these parameters, this algorithm is able to segment out most boundaries of images in the MSRC or iCoseg dataset. These initial segmentation based superpixels are sufficient for our experiments. We then extract the average color intensities of superpixels and make the 3D histograms with 20 bins in each dimension of the RGB color space.

Unless mentioned otherwise, we use the parameter settings of our approach as follows: $K = 6$, $\lambda = 1e-4$, $\lambda_0 = 1e5$, $\lambda_1 = 1e3$, $\lambda_2 = 1e-7$, and $T_{max} = 5$. There are some notations for selecting these parameters. The parameter K can be set as a small value (less than 10), since we only use these $2K$ GMM guide scribbles for initialization. If the value of K is large, it may produce some error labels caused by GMM. As mentioned above, we should keep a small λ which is a pre-condition of our local smooth energy, and keep a large λ_0 which guarantees the accuracy of the user scribbles. The parameter λ_1 controls the GMM guide scribbles, which should be less than λ_0. This setting is able to guarantee the accuracy of the user scribbles when the user and the GMM give different labels to the same superpixel. Then λ_2 should be set to a small value to balance the three energy terms, since the value of inter-image energy is much larger than the other two energies.

Fig. 5. Co-segmentation results on the panda1 set of images from iCoseg. The first column: input images. The second column: results by our algorithm without the inter-image energy. The third column: co-segmentation results by our full algorithm with the inter-image energy. The last column: ground truth. We run our algorithm on all 24 images and select 3 representative images for illustrating the advantage of the inter-image energy.

B. Analysis of Inter-Image Energy

We demonstrate the effectiveness of inter-image energy by running our algorithm on the panda1 image group (from iCoseg) with and without the inter-image energy in our total energy function. Since this algorithm without inter-image energy only includes the local smooth energy (LSE) and the global scribbled energy, we abbreviate this algorithm as LSE-GMM, which is the initialization method of y_i in Section III.E. Our full co-segmentation approach is then abbreviated as CHO (**Algorithm 1**). This panda1 group includes 24 images with a common object panda and the different complex backgrounds. Both of these algorithms perform well in most images in this group. The average precisions are 96.8% (LSE-GMM) and 98.2% (CHO), and the average Jaccard similarities are 92.4% (LSE-GMM) and 95.8% (CHO). Therefore, CHO outperforms LSE-GMM (+1.4% precision, +3.4% Jarccard similarity) in quantitative comparison.

In qualitative comparison, CHO also outperforms LSE-GMM. We select nine representative images (Fig. 5) for the intuitive comparison. For some images, LSE-GMM loses some foreground regions or produces some redundant foreground regions. The second column in Fig. 5 shows these problems of LSE-GMM. The images in all rows show that some foreground regions are lost. Almost all results by LSE-GMM contain redundant foreground regions except for the third row. The third and fourth columns in Fig. 5 show the results of CHO and the ground truth. These two groups of segmentation mask images are almost the same, which indicates the outstanding performance of CHO in these images.

The above problems of LSE-GMM may be due to the next two reasons. The first one is because of the incorrect guiding scribbles. They are automatically produced by GMM, but GMM does not work well in some images where the foreground and background are extremely similar. Fig. 6 shows examples of such incorrect guiding scribbles. In Fig. 6 (b) and (d), the red superpixels in the yellow rectangles indicate the incorrect guiding scribbles.

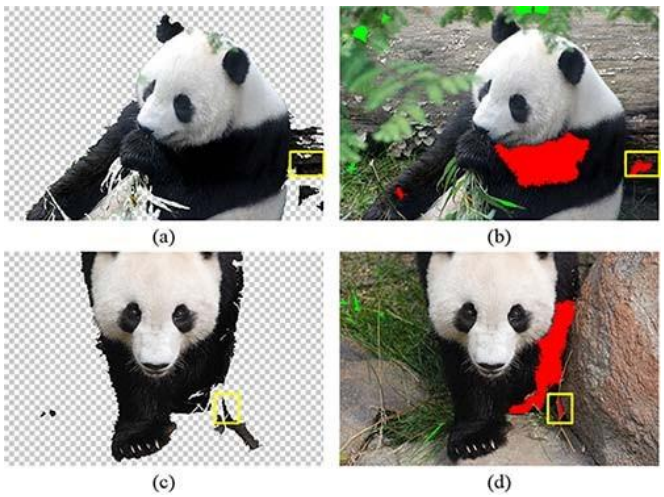


Fig. 6. Analyzing the incorrect guiding scribbles of LSE-GMM. (a) and (c) the segmentation results of LSE-GMM. (b) and (d) the labeled guiding scribbles by the global GMM, where the red regions indicate foreground and the green ones indicate background. The red labeled superpixels in the yellow rectangles of (b) and (d) are the wrong guiding scribbles. These incorrect guiding scribbles may appear in the regions where the background regions are extremely similar to foregrounds.

These guiding scribbles may lead to more incorrect segmentation regions in their neighborhoods shown in the yellow rectangles of Fig. 6 (a) and (c). The other reason is that these images lack enough foreground object information. The GMM only labels the superpixels with highest posterior probability, which may lose some foreground object information. Thus, we add the histogram constraint to overcome these problems and the results in the third column of Fig. 5 show the performance with this inter-image energy item. In summary, our full co-segmentation approach (CHO) can automatically correct the incorrect guiding scribbles. It also offers more foreground information to each image for improving the final co-segmentation performance.

C. Comparison With Single-Image Segmentation

Our approach of interactive co-segmentation for multiple images can also be viewed as a natural extension of the single-image segmentation via local spline regression (LSR) [22]. Naturally, we compare their method with our approach to verify the effectiveness. Furthermore, we also compare with other interactive single-image segmentation methods [7], [16], [33]. We randomly select three images as shown in Fig. 7 (a) from the MSRC dataset for experiments. There are similar backgrounds in these images, and some background regions are similar to the foreground object in a single image, such as the road regions in the first row of Fig. 7 (a). We directly run the source codes released by the authors to conduct their experiments with the recommended parameters. As shown in Fig. 7, we present the qualitative results between our approach and the other methods [7], [16], [22], [33], where our approach outperforms others using the same user scribbles. For example, LSR [22] generated unsatisfying boundary localization such as the second and third rows and produced some small noise regions

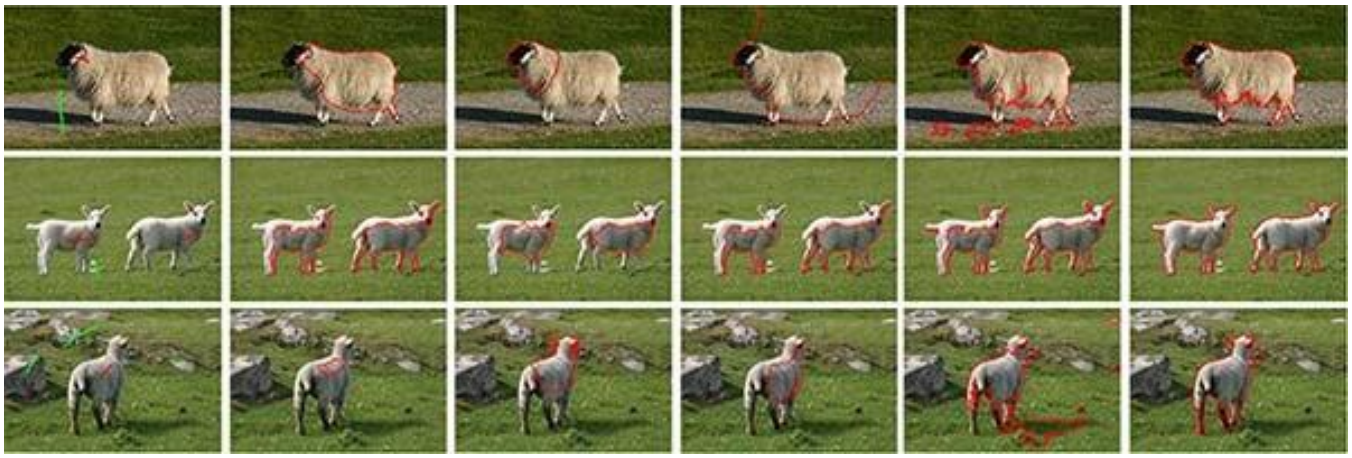


Fig. 7. Comparison results. (a) Input images with the same user scribbles. (b)-(e) Results by interactive single-image segmentation methods [7], [16], [22], and [33], respectively. (f) Results by our approach.

Fig. 8. Comparison results of [28], [29], and [36], and our co-segmentation approach. There are four class blocks in this figure. The first four classes: bear2, and Christ are from iCoseg. The last two classes: dog, and chair are from the MSRC dataset. In each block, the first row is the input images; the second, third and fourth rows are the results by the methods in [28], [29], and [36], respectively. The last row denotes the results by our approach.

like the first row in Fig. 7 (e). This is due to the lack of global knowledge of the foreground and the background in the other images. As shown in Fig. 7 (f), our approach leverages this information, and provides better co-segmentation performance.

D. Comparison With Other Co-Segmentation Methods

In this experiment, we used all previously selected image groups (30 groups) to evaluate our method. As shown in Fig. 8, the example results include four groups where

two groups are from iCoseg, and the other two groups are from MSRC. We selected five representative images from each class to illustrate the results. For example, in the bear group, the foreground object and parts of the background are similar. Some classes have similar backgrounds such as Christ, while the size and color appearance of a common object are different in some classes such as dog and chair. Our approach successfully segmented the common objects from these images. Our co-segmentation results are shown in the last row of each block. We also compared our approach with the previous unsupervised co-segmentation methods in [28] and [36] and the interactive so-segmentation method in [29]. The results of the unsupervised joint co-segmentation methods are offered by the authors in [36], who release their segmentation results of their method and several existing co-segmentation methods in their website. For the interactive co-segmentation method using random walks [29], we directly used the original implementations released by the authors to obtain the co-segmentation results. As shown in Fig. 8, these three well-known methods perform well in some images but not so accurate in others. However, some of the background pixels are always incorrectly segmented into the foreground objects by these approaches. In contrast, our approach achieved more accurate segmentation results than the compared methods with same scribbles, since our approach has strong self-adaptability for complex natural images. In the following, we will analyze these three methods to explain the detailed reasons.

As mentioned before, Joulin *et al.* [28] proposed a co-segmentation method by combining the spectral clustering and the discriminative clustering. The main idea is to find a classifier which is able to maximally separate the pixels of all images into k (here, we set $k = 2$) classes. This classifier works well for the images that have small variability in background, such as the Christ. But when the background and foreground are similar or the background has large variability (such as bear), it will fail to segment out the object. Moreover, this method performed not well for the classes with large variability in foreground, such as dog, and chair. This is due to that the classifier cannot classify these foregrounds as a class. Rubinstein *et al.* [36] proposed a joint object co-segmentation method by combining the saliency and the SIFT flow. The key insight of this method is that a common object should be salient within each image. Therefore it performed well in the images with high saliency, such as the second images. However, the parts of background, such as the people in the last image of the dog class, are also salient in some images, which lead this method to get incorrect segmentation results. Moreover, it cannot find the background regions which are inside the foreground, like the chair class, since these regions are salient in the images. Collins *et al.* [29] proposed an interactive co-segmentation method using random walks. Their method can segment out most background regions which are similar to the foreground in visual features but are different in semantic features such as the stone balustrade in the third image of the Christ group. These regions are difficult to be segmented out by unsupervised co-segmentation methods. However, this method also provided incorrect segmentation boundaries for some complicated images, such as bear, dog,

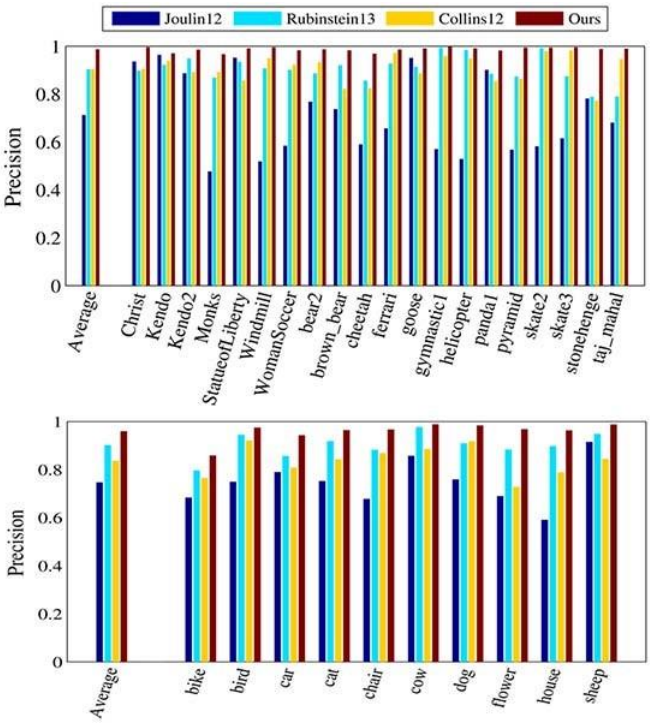


Fig. 9. Comparison results on iCoseg (top) and MSRC (bottom) between our method and other methods: Joulin12 [28], Rubinstein13 [36], and Collins12 [29]. The ratios of correctly labeled pixels (both foreground and background) are measured, and each plot shows the average per-class precision.

TABLE I
AVERAGE PRECISION ($\overline{P}$) AND JACCARD SIMILARITY ($\overline{J}$)

Method	iCoseg		MSRC	
	$\overline{P}$	$\overline{J}$	$\overline{P}$	$\overline{J}$
Joulin *et al.* [28]	71.2	46.1	74.7	49.8
Rubinstein *et al.* [36]	90.3	70.5	90.2	72.5
Collins *et al.* [29]	90.4	63.6	83.7	47.4
Ours	**98.7**	**93.1**	**96.0**	**85.5**

and chair. This may be due to that the united parameters are not suitable for all image classes. Thus, the unsuitable parameters may produce unexpected co-segmentation results. This method requires carefully tuning their parameters or tediously adding more scribbles to improve the co-segmentation results.

Furthermore, we quantitatively compare these above methods with our approach. Fig. 9 shows the per-class precision on iCoseg (top) and MSRC (bottom). According to this figure, the precision of our approach is higher than that of other three methods in each class. The average precision of our approach on iCoseg (98.7%) is more than the one on MSRC (96.0%), which is due to that the images with high variability in MSRC are more complex than those in iCoseg. The complexity of images in groups may decrease the performance of our approach, especially on the bike class. Table I shows the accurate average precision and Jaccard similarity on iCoseg and MSRC. It appears that our approach has significant improvement over the previously proposed methods.

E. Run Time

In this section, we depict the advantages of our approach for co-segmenting the images in terms of

TABLE II
AVERAGE RUN TIME (SECONDS) ON ICOSEG (FIRST 20 ROWS) AND MSRC (LAST 10 ROWS)

Datasets	images	Ours	[28]	[29]
Christ	13	3.7	716.0	6678.0
Kendo	30	9.7	3087.5	13361.4
Kendo2	11	2.1	561.7	5730.6
Monks	17	16.0	1047.0	7023.9
StatueofLiberty	41	7.6	3236.0	23316.1
Windmill	18	3.1	1165.0	10530.5
WomanSoccer	36	26.4	3158.2	18419.6
bear2	19	12.9	1391.2	7326.9
brown_bear	5	2.5	241.0	1782.8
cheetah	33	65.9	2315.1	14218.3
ferrari	11	8.1	759.5	5414.5
goose	31	7.9	1865.8	13350.7
gymnastic1	6	3.4	400.5	3947.3
helicopter	12	1.8	711.8	5229.7
panda1	24	24.8	1875.7	11483.6
pyramid	10	1.8	446.8	5102.3
skate2	12	2.1	724.7	6805.1
skate3	13	4.7	653.1	6310.7
stonehenge	18	4.4	1176.8	9971.0
taj_mahal	10	3.4	579.2	8035.9
Average	18.5	**10.6**	1305.6	9201.9
bike	30	12.8	2864.6	3476.3
bird	32	6.4	2671.3	3796.2
car	30	10.7	2130.0	3243.9
cat	24	4.6	1582.6	2932.7
chair	30	7.5	1654.3	3399.6
cow	30	7.5	2129.2	3221.4
dog	30	6.9	2215.5	3648.6
flower	32	13.7	4379.6	3723.1
house	30	7.9	1799.3	3408.1
sheep	30	6.5	1895.7	3038.8
Average	29.8	**8.5**	2332.2	3388.9

TABLE III
RUN TIME (SECONDS) ON LARGE-SCALE GROUPS OF IMAGES

Datasets	images	Ours	[24]
Car	200	66.2	370.8
Horse	200	68.4	345.3
Airplane	200	37.5	320.3
Average	200	**57.4**	345.5

computational efficiency. All the run time values in Tables II and III are measured in seconds on a Dell Precision T5600 workstation with Intel Xeon Processor E5-2680 CPU and 64GB RAM. We first run our approach and the other two co-segmentation methods [28], [29] on all previously selected image groups MSRC. The authors of [29] have provided the CPLEX solver based source code, where the memory requirement increases sharply with large-scale groups of images. Due to the limited memory on our workstation, we divided the big groups into subgroups (8-10 images) for running their code. We run the method in [28] and our approach on a whole group. The average run time is shown in Table II, which demonstrates that our approach is much faster than the other two methods.

Further, we evaluate the computational efficiency of our approach on the large-scale image groups (car, horse, and airplane). We compare with the other large-scale co-segmentation method [24] instead of the above two methods without scalability (i.e., the memory requirement of them is beyond our computation capability). Table III shows the statistics of run time. Our approach required 57.4 seconds to co-segment 200 images in average, which was much faster than the method in [24]. Especially for the airplane images, we only required 37.5 seconds. This may be due to that the sky background was over-segmented to fewer superpixels. Moreover, our approach also outperforms their method in both average precision and Jaccard similarity, which are shown in Table IV.

TABLE IV
AVERAGE PRECISION ($\overline{P}$) AND JACCARD SIMILARITY ($\overline{J}$) ON LARGE-SCALE IMAGE GROUPS

Method	Car		Horse		Airplane	
	$\overline{P}$	$\overline{J}$	$\overline{P}$	$\overline{J}$	$\overline{P}$	$\overline{J}$
Kim *et al.* [24]	66.1	9.5	81.7	2.2	85.1	4.1
Ours	**90.0**	**75.4**	**93.4**	**68.5**	**91.7**	**61.1**

V. CONCLUSION

In this work, we have presented a new framework for solving the interactive co-segmentation problem based on energy optimization. The proposed energy function consists of the global energy and the local energy. Our global energy successfully captures the information of user scribbles and the common foreground object in all related images. Our local energy is based on spline regression with adaptability to the complex natural images. An efficient iterative optimization algorithm is proposed to solve the proposed energy function for computation efficiency, which is able to process large-scale image sets. The experimental results have shown that our approach outperforms the previous co-segmentation methods by both quantitative and qualitative performance measurements. In future work, we will extend this framework to multi-class image or video co-segmentation [35], where the spatial-temporal coherence should be considered carefully.

APPENDIX: IMPLEMENTATION DETAILS

In this section, we describe our exact formulations of spline regression with more detailed explanations for completeness.

A. Details of Spline Regression

This sub-section discusses how to transform (4) to (5). Firstly, we recall up (4):

$$J(g_i) = \sum_{j=1}^{k_i} (z_{ij} - g_i(c_{ij}))^2 + \lambda S(g_i).$$

According to [1], we use a semi-norm to define $S(g_i)$. Then the minimizer g_i will be given by

$$g_i(c) = \sum_{j=1}^{d} \beta_{i,j} p_j(c) + \sum_{j=1}^{k_i} a_{i,j} \varphi_{i,j}(c), \qquad (23)$$

where $t = C_{d+s-1}^{s-1} = (d+s-1)!/(d!(s-1)!)$ and $\{p_j(c)\}_{j=1}^{t}$ are a set of primitive polynomials which can span the polynomial space with a degree less than s. Here s is the order of the partial derivatives in the semi-norm [2]. $a_i = [a_{i,1}, a_{i,2}, ..., a_{i,k_i}]^T \in R^{k_i}$ and $\beta_i = [\beta_{i,1}, \beta_{i,2}, ..., \beta_{i,t}]^T \in R^t$ are the coefficients concerning

superpixel c_i. $\varphi_{i,j}(c)$ is a Green function [1], which is defined by: $\varphi_{i,j}(c) = (c - c_{ij})^{2s-d}\log(c - c_{ij})$, if d is even; $\varphi_{i,j}(c) = (c - c_{ij})^{2s-d}$, if d is odd.

There are at most k_i equations by substituting the k_i features in the neighborhood N_i into (23). Thus, t new equations are introduced [22] to solve $k_i + t$ coefficients: $P_{i}a_i = 0$, where $P_i \in R^{t\times k_i}$ collects the values of the primitive polynomials $p_j(c)$ with the k_i features in neighborhood N_i as input. For example, when $d = 3$ and $s = 2$, t equals to 4 and c is denoted as $c = [c^{(1)}, c^{(2)}, c^{(3)}]^\mathsf{T}$. Then the primitive polynomials will be $p_1(c) = 1$, $p_2(c) = c^{(1)}$, $p_3(c) = c^{(2)}$, $p_4(c) = c^{(3)}$. We only use the conditions: $d = 3$ and $s = 2$ for computational simplicity in our approach.

Combining the k_i equations derived from (23), we have

$$\begin{bmatrix} K_i + \lambda I & P_i \\ P_i^\mathsf{T} & 0 \end{bmatrix} \begin{bmatrix} a_i \\ \beta_i \end{bmatrix} = \begin{bmatrix} Z_i \\ 0 \end{bmatrix}, \tag{24}$$

where K_i is a $k_i \times k_i$ symmetrical matrix with rr th row and ccth column element $K_{rr,cc} = \varphi_{i,cc}(c_{irr})$. $Z_i = [z_{i_1}, z_{i_2}, \ldots, z_{i_{k_i}}]^\mathsf{T} \in R^{k_i}$ collects the labels of the k_i superpixels in N_i, and I is a $k_i \times k_i$ identity matrix.

According to [2], the regularized loss minimization $J(g_i(c))$ can be approximately evaluated in (4):

$$J(g_i) \approx \sum_{j=1}^{k_i} (z_{i_j} - g_i(c_{i_j}))^2 + \lambda a_i^\mathsf{T} K_i a_i. \tag{25}$$

According to (24), the conditions $z_{i_j} = g_i(c_{i_j})$, $j = 1, 2, \ldots, k_i$ can be approximately satisfied, when λ is small enough. This indicates that the first term in (4) can be treated as zeros. Then, we have $J(g_i) \approx \lambda a_i^\mathsf{T} K_i a_i$.

Theorem 1: For a small λ the regularized loss minimization can be evaluated in terms of class label vector:

$$J(g_i) \approx \lambda Z_i^\mathsf{T} M_i Z_i, \tag{26}$$

where M_i is the upper left $k_i \times k_i$ submatrix of the inverse matrix of the coefficient matrix in (24).

Proof: Based on (24), we have $(K_i + \lambda I) * a_i + P_i\beta_i = Z_i$, $P_i^\mathsf{T} * a_i = 0$. Note that M_i is the upper left $k_i \times k_i$ submatrix of the inverse matrix of the coefficient matrix in (24). Then, the solution of (24) includes $a_i = M_i Z_i$.

It follows that $a_i^\mathsf{T}(K_i + \lambda I) * a_i + a_i^\mathsf{T} P_i^\mathsf{T}\beta_i = a_i^\mathsf{T} Z_i \Longrightarrow a_i^\mathsf{T}(K_i + \lambda I) * a_i = a_i^\mathsf{T} Z_i M_i Z_i$. The last equality holds since M_i is a symmetrical matrix. Thus, for a small λ, we have $J(g_i) \approx \lambda a_i^\mathsf{T} K_i * a_i \approx a_i^\mathsf{T} K_i a_i + \lambda^2 a_i^\mathsf{T} a_i = Z_i^\mathsf{T} M_i Z_i$.

B. Invertibility of L_i^r in (21) or (22)

Proof: We first introduce some properties of L_i^r and some denotations. Since the matrix M_i^{smooth} is a Laplacian matrix [22], M_i^{smooth} is positive semidefinite. In other words, for any nonzero vector $y_i \in R^{m_i}$, $y_i^\mathsf{T} M_i^{smooth} y_i \geq 0$. Denote $\{\tilde{\lambda}_{i1}, \tilde{\lambda}_{i2}, \ldots, \tilde{\lambda}_{im_i}\}$ and $\{u_{i1}, u_{i2}, \ldots, u_{im_i}\}$ as eigenvalues and eigenvectors of L_i^r. Without loss of generality, we assume $\tilde{\lambda}_{i1} \leq \tilde{\lambda}_{i2} \leq \cdots \leq \tilde{\lambda}_{im_i}$. Then we have $\tilde{\lambda}_{i1} = 0$, $\tilde{\lambda}_{ij} > 0$, $j = 2, \ldots, m_i$, $u_{i1} = \mathbf{1}$, where $\mathbf{1}$ is a full one vector. For a connected graph, the Laplacian matrix M_i^{smooth} has a single eigenvalue corresponding to zero, and the corresponding eigenvector is constant [3]. Since $M_i^{smooth}\mathbf{1} = 0$, $u_{i1} = \mathbf{1}$.

For simplicity, we denote $L_i^r = M_i^{smooth} + \tilde{D}_i$, where $\tilde{D}_i = \lambda_1 D_i + D_i^s$, $i = 1, 2, \ldots, l$, $\tilde{D}_i = \lambda_1 D_i$, $i = l + 1, l + 2, \ldots, m_i$. It is easy to find that $y_i^\mathsf{T} \tilde{D}_i y_i \geq 0$, for $i = 1, 2, \ldots, m_i$, since $\tilde{D}_i$ is a diagonal matrix and the diagonal elements are nonnegative.

We show the matrix L_i^r is positive definite, i.e. for any nonzero vector $y_i \in R^{m_i}$, $y_i^\mathsf{T} L_i y_i > 0$. We discuss the following two situations:

1. $y_i^\mathsf{T} D_i y_i > 0$. Under this situation, $y_i^\mathsf{T} L_i y_i = y_i^\mathsf{T} M_i^{smooth} y_i + y_i^\mathsf{T} D_i y_i > 0$, since $y_i^\mathsf{T} M_i^{smooth} y_i \geq 0$.
2. $y_i^\mathsf{T} D_i y_i = 0$. We can set $y_i = \sum_{j=1}^{m_i} \beta_{ij} u_{ij}$, since the eigenvectors are linearly independent.

According to spectral decomposition of a matrix, we have $M_i^{smooth} = \sum_{j=1}^{m_i} \tilde{\lambda}_{ij} u_{ij} u_{ij}^\mathsf{T}$. Then, $y_i^\mathsf{T} M_i^{smooth} y_i = \sum_{j=1}^{m_i} \beta_{ij}\tilde{\lambda}_{ij}(u_{ij}^\mathsf{T} u_{ij})(u_{ij}^\mathsf{T} u_{ij})$, since the eigenvectors are orthogonal. If $y_i^\mathsf{T} M_i^{smooth} y_i = 0$, then for $j = 2, \ldots, m_i$, $\beta_{ij} = 0$, since $\tilde{\lambda}_{ij} > 0$, $u_{ij}^\mathsf{T} u_{ij} > 0$. So $y_i = \beta_{i1} u_{i1} = \beta_{i1}\mathbf{1}$. However $y_i^\mathsf{T} \tilde{D}_i y_i = \beta_{i1}^2 \mathbf{1}^\mathsf{T} \tilde{D}_i \mathbf{1} > 0$, which violates the condition $y_i^\mathsf{T} \tilde{D}_i y_i = 0$. Thus, $y_i^\mathsf{T} M_i^{smooth} y_i > 0$. Then $y_i^\mathsf{T} L_i y_i = y_i^\mathsf{T} M_i^{smooth} y_i + y_i^\mathsf{T} \tilde{D}_i y_i > 0$. For any nonzero vector $y_i \in R^{m_i}$, we have $y_i^\mathsf{T} L_i y_i > 0$, i.e. the matrix L_i^r is positive definite and invertible.

C. Convergence of the Iterative Optimization

Proof: The energy function $E(\tilde{Y})$ in (14) corresponds to a bound-constrained QP problem. Obviously, this energy function is a convex function. So we only need to guarantee the energy does not go up at each iteration, i.e., $E(\tilde{Y}_0) \geq E(Y_1) \geq E(Y_2) \geq \cdots \geq E(Y^*)$, where Y_t is the solution at the $t-th$ iteration, $\tilde{Y}_0$ is the initial solution, and $\tilde{Y}^*$ is the optimizing solution. Then the iterative optimization algorithm will converge. Note that we process each image with a coordinate sequence at each iteration, and we then perform the following two steps for each image. The first step is to optimize the sub-problem in (16), and the second step is to update the global foreground histogram. Then we can denote $E_{ti}(y_i)$ and $\hat{E}_{ti}(\bar{h})$ as the total energy after two steps for image i at t iteration.

Note that y_j, $j = 1, 2, \ldots, n$, $\hat{E}_{ti}(\bar{h})$ will be the optimizing value of $E(\tilde{Y})$. Then we have $E(y) \geq \hat{E}_{ti}(\bar{h})$. Similarly, we can get $\hat{E}_{ti}(\bar{h}) \geq E_{t(i+1)}(y_{i+1})$. Then we have $E_{t(i+1)}(y_{i+1}) \geq E_{ti}(y_i)$. According to our iterative steps, it is easy to find that $E(\tilde{Y}_{t-1}) = \hat{E}_{(t-1)n}(\bar{h}) \geq E_{t1}(y_1) \geq E_{t2}(y_2) \geq \cdots \geq E_{tn}(y_n) \geq \hat{E}_{tn} = E(\tilde{Y}_t)$, i.e. $E(\tilde{Y}_{t-1}) \geq E(\tilde{Y}_t)$ for $t = 1, 2, \cdots$.

REFERENCES

[1] J. Duchon, "Splines minimizing rotation-invariant semi-norms in Sobolev spaces," in *Constructive Theory of Functions of Several Variables* (Lecture Notes in Mathematics), vol. 571. Berlin, Germany: Springer-Verlag, 1977, pp. 85–100.

[2] G. Wahba, *Spline Models for Observational Data*. Philadelphia, PA, USA: SIAM, 1990.

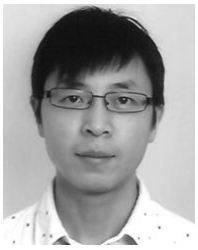
[3] N. Biggs, "Algebraic potential theory on graphs," *Bull. London Math. Soc.*, vol. 29, no. 6, pp. 641–682, 1997.

[4] L. Zhang and Q. Ji, "A Bayesian network model for automatic and interactive image segmentation," *IEEE Trans. Image Process.*, vol. 20, no. 9, pp. 2582–2593, Sep. 2011.

[5] P. E. Gill, W. Murray, and M. H. Wright, *Numerical Linear Algebra and Optimization*. New York, NY, USA: Perseus Books, 1990.
[6] J. Shen, Y. Du, W. Wang, and X. Li, "Lazy random walks for superpixel segmentation," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1451–1462, Apr. 2014.
[7] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proc. IEEE ICCV*, Jul. 2001, pp. 105–112.
[8] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.
[9] C. Rother, A. Blake, and V. Kolmogorov, "'GrabCut': Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.
[10] N. Gould and P. L. Toint, "Preprocessing for quadratic programming," *Math. Program.*, vol. 100, no. 1, pp. 95–132, 2004.
[11] J. Winn, A. Criminisi, and T. Minka, "Object categorization by learned universal visual dictionary," in *Proc. 10th IEEE ICCV*, Oct. 2005, pp. 1800–1807.
[12] C. Rother, T. Minka, A. Blake, and V. Kolmogorov, "Cosegmentation of image pairs by histogram matching—Incorporating a global constraint into MRFs," in *Proc. IEEE CVPR*, Jun. 2006, pp. 993–1000.
[13] Y. Mu and B. Zhou, "Co-segmentation of image pairs with quadratic global constraint in MRFs," in *Proc. ACCV*, 2007, pp. 837–846.
[14] A. K. Sinop and L. Grady, "A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm," in *Proc. IEEE ICCV*, Oct. 2007, pp. 1–8.
[15] S. Calinon, F. Guenter, and A. Billard, "On learning, representing, and generalizing a task in a humanoid robot," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 2, pp. 286–298, Apr. 2007.
[16] A. Criminisi, T. Sharp, and A. Blake, "GeoS: Geodesic image segmentation," in *Proc. ECCV*, 2008, pp. 99–112.
[17] L. Mukherjee, V. Singh, and C. R. Dyer, "Half-integrality based algorithms for cosegmentation of images," in *Proc. IEEE CVPR*, Jun. 2009, pp. 2028–2035.
[18] D. S. Hochbaum and V. Singh, "An efficient algorithm for co-segmentation," in *Proc. IEEE ICCV*, Sep./Oct. 2009, pp. 269–276.
[19] A. Joulin, F. Bach, and J. Ponce, "Discriminative clustering for image co-segmentation," in *Proc. IEEE CVPR*, Jun. 2010, pp. 1943–1950.
[20] S. Vicente, V. Kolmogorov, and C. Rother, "Cosegmentation revisited: Models and optimization," in *Proc. ECCV*, 2010, pp. 465–479.
[21] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, "iCoseg: Interactive co-segmentation with intelligent scribble guidance," in *Proc. IEEE CVPR*, Jun. 2010, pp. 3169–3176.
[22] S. Xiang, F. Nie, and C. Zhang, "Semi-supervised classification via local spline regression," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 2039–2053, Nov. 2010.
[23] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, "Interactively co-segmentating topically related images with intelligent scribble guidance," *Int. J. Comput. Vis.*, vol. 93, no. 3, pp. 273–292, 2011.
[24] G. Kim, E. P. Xing, L. Fei-Fei, and T. Kanade, "Distributed cosegmentation via submodular optimization on anisotropic diffusion," in *Proc. IEEE ICCV*, Nov. 2011, pp. 169–176.
[25] S. Vicente, C. Rother, and V. Kolmogorov, "Object cosegmentation," in *Proc. IEEE CVPR*, Jun. 2011, pp. 2217–2224.
[26] L. Mukherjee, V. Singh, and J. Peng, "Scale invariant cosegmentation for image groups," in *Proc. IEEE CVPR*, Jun. 2011, pp. 1881–1888.
[27] 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 CVPR*, Jun. 2011, pp. 2129–2136.
[28] A. Joulin, F. Bach, and J. Ponce, "Multi-class cosegmentation," in *Proc. IEEE CVPR*, Jun. 2012, pp. 542–549.
[29] 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 CVPR*, Jun. 2012, pp. 1656–1663.
[30] L. Mukherjee, V. Singh, J. Xu, and M. D. Collins, "Analyzing the subspace structure of related images: Concurrent segmentation of image sets," in *Proc. ECCV*, 2012, pp. 128–142.
[31] J. C. Rubio, J. Serrat, A. Lopez, and N. Paragios, "Unsupervised co-segmentation through region matching," in *Proc. IEEE CVPR*,unsupervised energy minimization Jun. 2012, pp. 749–756.
[32] E. Kim, H. Li, and X. Huang, "A hierarchical image clustering cosegmentation framework," in *Proc. IEEE CVPR*, Jun. 2012, pp. 686–693.
[33] T. N. A. Nguyen, J. Cai, J. Zhang, and J. Zheng, "Robust interactive image segmentation using convex active contours," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3734–3743, Aug. 2012.
[34] X. Dong, J. Shen, and L. Van Gool, "Segmentation using subMarkov random walk," in *Proc. EMMCVPR*, 2015, pp. 237–248.
[35] W. Wang, J. Shen, X. Li, and F. Porikli, "Robust video object cosegmentation," *IEEE Trans. Image Process.*, vol. 24, no. 10, pp. 3137–3148, Oct. 2015.
[36] M. Rubinstein, A. Joulin, J. Kopf, and C. Liu, "Unsupervised joint object discovery and Segmentation in Internet images," in *Proc. IEEE CVPR*, Jun. 2013, pp. 1939–1946.

Xingping Dong is currently pursuing the Ph.D. degree with the School of Computer Science, Beijing Institute of Technology, Beijing, China. His current research interests include random walks and image segmentation.

Jianbing Shen (M'11–SM'12) is currently a Professor with the School of Computer Science, Beijing Institute of Technology, Beijing, China. His research interests include computer vision and multimedia processing. He has received many flagship honors, including the Fok Ying Tung Education Foundation from the Ministry of Education, the Program for Beijing Excellent Youth Talents from the Beijing Municipal Education Commission, and the Program for New Century Excellent Talents from the Ministry of Education. He is on the Editorial Boards of *Neurocomputing*.



Ling Shao (M'09–SM'10) is currently a Full Professor and the Head of the Computer Vision and Artificial Intelligence Group with the Department of Computer Science and Digital Technologies, Northumbria University, Newcastle upon Tyne, and an Advanced Visiting Fellow with the Department of Electronic and Electrical Engineering, University of Sheffield. His research interests include computer vision, image processing, pattern recognition, and machine learning. He is a fellow of the British Computer Society and the Institution of Engineering and Technology, and a Life Member of the Association for Computing Machinery. He is an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON CYBERNETICS, and other journals.

Ming-Hsuan Yang (M'00-SM'06) received the Ph.D. degree in computer science from the University of Illinois at Urbana–Champaign, in 2000. He is currently an Associate Professor of Electrical Engineering and Computer Science with the University of California, Merced. He is a Senior Member of the Association for Computing Machinery. He received the NSF CAREER Award in 2012, the Senate Award for Distinguished Early Career Research at UC Merced in 2011, and the Google Faculty Award in 2009. He served as an Associate Editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE from 2007 to 2011, and is an Associate Editor of the *International Journal of Computer Vision*, *Image and Vision Computing*, and the *Journal of Artificial Intelligence Research*.