Multi-View Saliency-Guided Clustering for Image Cosegmentation

Zhiqiang Tao, Hongfu Liu, *Member, IEEE*, Huazhu Fu, *Senior Member, IEEE*, and Yun Fu, *Fellow, IEEE*

***Abstract*— Image cosegmentation aims at extracting the common objects from multiple images simultaneously. Existing methods mainly solve cosegmentation via the pre-defined graph, which lacks flexibility and robustness to handle various visual patterns. Besides, similar backgrounds also confuse the identification of the common foreground. To address these issues, we propose a novel multi-view saliency-guided clustering algorithm (MvSGC) for the image cosegmentation task. In our model, the unsupervised saliency prior is used as partition-level side information to guide the foreground clustering process. To achieve robustness to noises and missing observations, similarities on an instance-level and the partition-level are both considered. Specifically, a unified clustering model with cosine similarity is proposed to capture the intrinsic structure of data and keep the partition result consistent with the side information. Moreover, we leverage multi-view weight learning to integrate multiple feature representations to further improve the robustness of our approach. A K-means-like optimization algorithm is developed to proceed the constrained clustering in a highly efficient way with theoretical support. The experimental results on three benchmark datasets (i.e., the iCoseg, MSRC, and Internet image dataset) and one RGB-D image dataset demonstrate the superiority of applying our clustering method for image cosegmentation.**

***Index Terms*— Image cosegmentation, constrained clustering, saliency prior, cosine similarity, multi-view learning.**

I. Introduction

IMAGE cosegmentation has drawn a great deal of research efforts in the computer vision community, which could be used in a wide range of applications such as interaction image editing [1], content-based image retrieval [2], and automatic image annotation [3]. Different from traditional single image segmentation, cosegmentation leverages the co-occurrence prior of similar foreground objects and aims to segment these common objects from multiple images simultaneously [4]–[6]. Recently, as the amount of photo sharing and multi-modality data (*e.g.*, RGB-D) grow rapidly, image data containing common semantics have become ubiquitous and diverse. Hence, a robust and efficient cosegmentation method, which is also flexible to handle various visual cues, is highly desired.

Existing cosegmentation methods could be roughly divided into two categories. The graph-based methods [4], [7], [8] formulate image elements (*e.g.*, superpixels or object proposals) as graph nodes, and transfer cosegmentation task into an instance selection problem. The performance of these methods heavily depends on the graph construction, which is usually sensitive to the edge definition, and thus, lack of robustness to the appearance variation. Moreover, since various priors and feature similarities are incorporated into the graph construction process, the graph-based method generally has a low flexibility to handle different visual data. On the other hand, the clustering-based method [5], [9], [10] solves cosegmentation by partitioning the common objects apart from background. Nevertheless, these previous works still perform clustering algorithms on a pre-defined graph, which may suffer from the high time complexity and also lack robustness and flexibility like the graph-based ones. These drawbacks obviously limit their applications. Different from existing methods, we target to develop a robust, flexible and highly efficient clustering algorithm for the image cosegmentation task.

However, it is not an easy task to cluster similar image elements into the same group, due to different illumination, viewpoint change, arbitrary poses and object deformation existing in natural images. It is difficult to identify the common foreground objects from co-occurring backgrounds among multiple images without giving any prior knowledge of foregrounds. In light of this, we come up with idea of using constrained clustering to improve cosegmentation performance with priori knowledge. Recently, Liu and Fu [11], Liu *et al.* [12] proposed an interesting clustering method that formulates given labels as a partition-level side information to facilitate the clustering process, which has shown promising performance for the constrained clustering. However, it is not straightforward to apply this method in image cosegmentation for two reasons: 1) Ref [11] is not applicable for unsupervised task as its side information is derived from ground-truth annotation; 2) its objective function is induced by the squared Euclidean distance with the categorical utility function [13], which does not perform well for high-dimensional and sparse image features (*e.g.*, bag-of-words).

To address the above issues, we propose a novel Multi-view Saliency-Guided Clustering algorithm (MvSGC) with cosine similarity for the image cosegmentation task.

Manuscript received June 29, 2018; revised December 17, 2018 and February 11, 2019; accepted April 9, 2019. Date of publication May 8, 2019; date of current version July 16, 2019. This work was supported in part by the NSF IIS Award under Grant 1651902. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Shuicheng Yan. *(Corresponding author: Zhiqiang Tao.)*

Z. Tao is with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115 USA (e-mail: zqtao@ece.neu.edu).

H. Liu is with the Michtom School of Computer Science, Brandeis University, Waltham, MA 02453 USA (e-mail: hongfuliu@brandeis.edu).

H. Fu is with the Inception Institute of Artificial Intelligence (IIAI), Abu Dhabi, United Arab Emirates (e-mail: huazhufu@gmail.com).

Y. Fu is with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115 USA, and also with the Khoury College of Computer and Information Sciences, Northeastern University, Boston, MA 02115 USA (e-mail: yunfu@ece.neu.edu).

Digital Object Identifier 10.1109/TIP.2019.2913555

1057-7149 © 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

The unsupervised saliency prior is formulated as a partition-level side information to give the prior knowledge of foreground and also suppress the common backgrounds. To enhance the robustness of our model, we alleviate noises in the given prior by jointly considering instance-level and partition-level similarity. Specifically, we calculate cosine similarity between data point and its cluster center; and measure the similarity between clustering result and side information with a cosine utility function. These two level similarities are involved by a unified clustering model upon an auxiliary matrix, where a K-means-like optimization is developed with a roughly linear time complexity. Moreover, we leverage a multi-view weight learning approach to further improve the robustness of MvSGC. The clustering process and weight learning are proceeded in an alternative optimization way. We summarize the contributions of this paper as follows.

1) We introduce a novel method of leveraging partition-level side information to guide the clustering process for image cosegmentation task (Section III-C). By jointly considering feature and partition similarities, our method exhibits a high robustness to the noises and missing observations in the side information, which makes it available for unsupervised prior.
2) The proposed MvSGC is built upon cosine similarity (Section III-D1), which could better handle non-spherical cluster structure than the Euclidean distance. Nontrivially, we provide theoretical supports to give a new insight for the cosine utility function, leading to a K-means-like optimization solution of high efficiency.
3) We give an adaptive multi-view weighting approach (Section III-D2) to utilize the complementary information among multiple views, including low-level appearance features and higher-level deep representations, which can further improve the robustness of our algorithm to capture various appearances in natural images.
4) Besides the experiments for image cosegmentation, we also test our cosegmentation method with RGB-D images by adding depth prior into the side information, which showcases the flexibility of MvSGC to handle different modality data (Section IV-D).

A preliminary version of this work has been reported in [14]. Compared with [14], some major differences in this paper are as following: (1) Multi-view weight learning process is incorporated into our constrained clustering model with an alternative optimization solution. (2) Higher-level information encoded by deep CNN features is utilized by our approach to further improve the performance. (3) More model discussions, theoretical analyses and experimental evaluations are provided. (4) RGB-D image cosegmentation is conducted to show the flexibility of our algorithm.

The remainder of this paper is organized as follows. Section II gives a brief introduction to the existing cosegmentation literature. In Section III, we elaborate the proposed MvSGC clustering algorithm and give its solution. Experiments conducted on three benchmark datasets and one RGB-D image dataset, along with extensive model discussions, are presented in Section IV. Finally, we give a conclusion in Section V.

II. Related Work

A. Graph-Based Cosegmentation

Graph-based cosegmentation utilizes a graph model to organize the instances from images, and selects common objects by optimizing the objective function defined with the graph. It can effectively use the corresponding information shared with images, but the edge of graph is difficult to define. Rother *et al.* [4] first introduced cosegmentation as to jointly segment common objects from an image pair, and proposed to solve it by histogram matching with a Markov Random Filed (MRF) framework. Inspired by this work, lots of research efforts have been made to solve cosegmentation via a MRF model, which generally design unary potential to capture object boundary and pairwise potential to force the consistency between foreground objects. These methods mainly achieve cosegmentation by solving an energy minimization problem (*i.e.*, MAP estimation on a MRF), such as half-integrality algorithms [15], max-flow min-cut optimization [7], and the scale-invariant model via rank constraint [16].

Besides the MRF model, some other interesting graph-based methods include object proposal selection [2], submodular optimization [17], graph/region matching [18], [19], and consistent functional maps [20]. Recently, Fu *et al.* [21] proposed an object-based cosegmentation method for both regular color images and RGB-D images, where they defined the unary term with saliency and depth information and solved the problem by integer quadratic program. Quan *et al.* [8] formulated cosegmentation as a graph-based manifold ranking problem. They utilized pseudo background prior to initialize seeds, developed a two-stage framework to refine the probability map, and obtained the final result by Grab-Cut [22].

Moreover, several important extensions upon image cosegmentation have been widely studied such as interactive cosegmentation [1], [23], [24], multiple foreground cosegmentation (MFC) [25], [26] and video object cosegmentation [27]–[29]. The interactive cosegmentation methods usually utilize user scribbles to enhance the appearance model of foreground/background objects, which is in essence a semi-supervised method. For example, Dong *et al.* [23] employed the user interaction information to compute a global energy for the foreground and background regions in their graph model. More recently, Yuan and Lu [30] provided an end-to-end deep network for image cosegmentation, which trains the network on an auxiliary dataset. MFC methods generalize traditional cosegmentation into the multi-class case, which tries to segment multiple common objects among images. To solve this problem, Ma *et al.* [26] proposed a ℓ_1-manifold hyper-graph graph to fully explore the coherence across difference images. Different from extracting objects, Meng *et al.* [31] targeted to achieve part-level object segmentation, which is analogous to a fine-grained MFC problem. Video cosegmentation has a tight link to image cosegmentation, where it aims to simultaneously extract common objects from multiple videos. Similar to image cosegmentation, video

cosegmentation methods mainly adopt energy minimization on a pre-defined graph, *e.g.*, the multi-state selection graph model [27] and the spatial-temporal auto-context model [29]. In this paper, we focus on the unsupervised image cosegmentation task.

B. Clustering-Based Cosegmentation

Clustering-based cosegmentation usually imposes a global constraint on the clustering process to utilize the correspondence information of common foregrounds across multiple images. Joulin *et al.* [5] introduced a discriminative clustering framework for image cosegmentation, and they further extended it to tackle with multi-class cosegmentation by combining spectral and discriminative clustering [9]. Kim *et al.* [32] solved cosegmentation via spectral graph partitioning, where images are described by hierarchical superpixel layers, and an affinity matrix is computed with intra-image and inter-image edges. Some clustering-based methods also solve image cosegmentation by adopting Random walk algorithm. For example, Collins *et al.* [33] performed image cosegmentation by developing the constraint to match feature histograms of foreground objects in the random walk process. Recently, Lee *et al.* [10] proposed a multiple random walkers (MRW) algorithm and devised a restart rule for the clustering task. They obtained the cosegmentation result via a two-step framework, consisting of inter-image concurrence computation and intra-image MRW clustering. Some other recent clustering based cosegmentation methods could be found as the affinity propagation clustering model based on shape conformability [34], the constrained graph clustering model for solving MFC [35], and the discriminative clustering model with object part detector [36]. Compared with these clustering-based methods, we provide a K-means-like clustering solution, which does not depend on a pre-defined graph model of images and enjoys a roughly linear time complexity.

Similar to us, previous works in [6], [21], [37] also employed saliency prior to guide cosegmentation, where they mainly formulated cosegmentation as a graph-based model and utilized saliency prior to define the unary potential. More recently, Jerripothula *et al.* [38] provided a saliency co-fusion framework to highlight the correspondence among images, and finally achieved cosegmentation by using Grab-Cut [22] with the fused saliency maps. Ren *et al.* [39] proposed a unified optimization model to alternatively proceed saliency prior learning and tree graph matching. They employed the learned saliency to iteratively boost the cosegmentation result. Different from all the above methods, our approach cosegments images via a constrained clustering framework, which leverages saliency prior as a partition-level side information to guide the clustering process, and thus provides a novel way of using saliency priors for image cosegmentation.

III. The Proposed Method

A. Overall Framework

The framework of using our MvSGC clustering algorithm for image cosegmentation is shown in Fig. 1. Given a group of images (a), we first conduct saliency detection algorithm to obtain a set of saliency maps (b), and over-segment each image as a set of superpixels (c) with the SLIC method [40] (some other superpixel methods, *e.g.*, [41], are also applicable). After that, based on the results in (b) and (c), all the superpixels are divided into three groups (*i.e.*, *background*, *foreground* and *missing*) to obtain the partition-level side information (d). As shown in (e), multi-view feature descriptors are extracted from superpixels, containing both low-level appearance features and high-level semantic representation. Finally, the side information (d) and multi-view features (e) are fed into the proposed MvSGC algorithm to achieve cosegmentation results (f) via a one-step clustering process, which partitions all the superpixels in (c) into two classes as foreground and background.

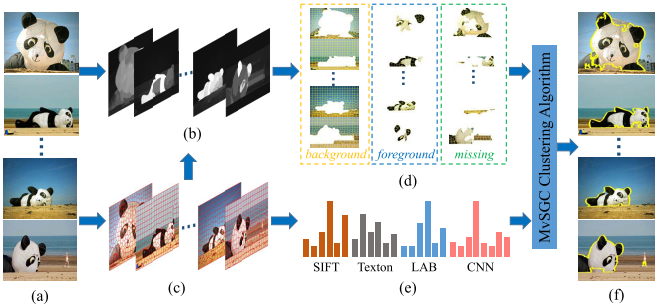


Fig. 1. Overview of the proposed MvSGC for image cosegmentation. (a) Image set. (b) Saliency priors. (c) Superpixels. (d) Partition-level side information. (e) Multi-view features. (f) Cosegmentation.

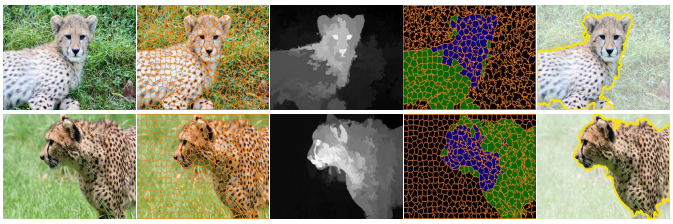
B. Saliency-Guided Partition-Level Side Information

Generally, using saliency prior for image cosegmentation has two main benefits: (1) saliency detection could identify the clues for foreground regions in an unsupervised and rapid way; (2) it effectively suppresses the misleading from common backgrounds by highlighting the salient object regions [42], [43]. However, the foreground prior given by saliency detection might be noisy and lead to incorrect results. Hence, a partial observation strategy is employed to formulate saliency prior as the partition-level side information.

Let $\mathcal{X}$ be a set of superpixels from all the images. Without loss of generality, we denote M as a saliency map of the image containing x for $\forall x \in \mathcal{X}$, and represent the saliency score of superpixel x as $M(x) \in [0, 1]$, which is calculated by averaging all the saliency values within x. Then, the side information S could be defined as

$$S(x) = \begin{cases} 2\text{: } \textit{foreground}, & M(x) \geq T_f \\ 1\text{: } \textit{background}, & M(x) \leq T_b \\ 0\text{: } \textit{missing}, & \text{otherwise}, \end{cases} \tag{1}$$

where T_f is a threshold for foreground and T_b for background ($T_b < T_f$). As following [44], T_f is set by the adaptive threshold as $T_f = \mu + \delta$, where μ and δ are corresponding to the mean value and standard deviation of M, respectively. Rather than directly assigning *background* to the remainder, we introduce $T_b = \mu$ as a pseudo background threshold, which regards the superpixels below the average saliency value as background regions. By using Eq. (1), the uncertainty of



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

Fig. 2. Illustration of the saliency-guided partition-level side information. (a) Image group. (b) Superpixels. (c) Saliency prior. (d) Visualization of the side information, where the labels of *background*, *foreground* and *missing* are colored by black, blue and green, respectively. (e) Our cosegmentation results.

saliency prior is remained as *missing* observations to avoid incorrectly labeling.

Fig. 2 gives an example of deriving partition-side information from saliency priors. For this case, only a part of foreground regions are correctly labeled as *foreground* (*i.e.*, the blue colored superpixels in Fig. 2(d)) by using the adaptive threshold. Thus, compared with simply regarding the remainder regions as *background*, our partial observation strategy can "save" a great deal of foreground regions into the *missing* group (*i.e.*, the green colored superpixels). It is worth noting that, most of these regions are finally segmented as foreground object in Fig. 2(e), which shows our clustering algorithm can alleviate the deficiency of saliency prior effectively.

C. The Proposed MvSGC Algorithm

Given a set of superpixels $\mathcal{X}$ from multiple images, we solve cosegmentation as a constrained clustering problem that targets to partition all the superpixels in $\mathcal{X}$ into $K = 2$ classes. We refer to n as $|\mathcal{X}|$. Let $X = [X^{(1)}, \cdots, X^{(r)}]$ be the multi-view feature matrix for $\mathcal{X}$, where r is the number of multiple views and $X^{(i)} \in \mathbb{R}^{n\times d^{(i)}}$ corresponds to the i^{th} view's feature of $d^{(i)}$ dimensionality, $1 \le i \le r$, and $S \in \mathbb{R}^{n\times 1}$ be the side information that partitions $\mathcal{X}$ into three groups. The objective function of our MvSGC clustering algorithm is given by

$$\min_{\pi, w_i} \sum_{k=1}^{K} \sum_{x\in\mathcal{C}_k} \sum_{i=1}^{r} w_i^{\gamma} f(x^{(i)}, m_k^{(i)}) - \lambda U_{cos}(\pi \otimes S, S)$$
$$\text{s.t. } \pi \in \{1, \ldots, K\}^{n\times 1}, \quad \sum_{i=1}^{r} w_i = 1, \ \forall i, w_i \ge 0, \qquad (2)$$

where π is the partition result assigning each superpixel a label in $\{1, \cdots, K\}$, w_i is the weight of the i^{th} view, x represents the superpixel described by $[x^{(1)}, \ldots, x^{(r)}]$, and $\mathcal{C}_k$ is the superpixel set of the k^{th} cluster in π. Two parameters $\gamma, \lambda > 0$ are used in our model, where γ controls the weight distribution among all the views as suggested by [45], and λ balances multi-view features and the side information. $x^{(i)}$ and $m_k^{(i)}$ represent one row in $X^{(i)}$ and the centroid of $\mathcal{C}_k$ in $X^{(i)}$, respectively. We denote $f(\cdot,\cdot)$ as the cosine distance, which is defined by

$$f(a, b) = \|a\|(1 - \frac{\langle a, b\rangle}{\|a\|\|b\|}) = \|a\|(1 - \cos(a, b)), \qquad (3)$$

where a, b are two vectors of the same size, $\|\cdot\|$ refers to the ℓ_2 norm, $\langle a, b\rangle$ denotes the inner product and $\cos(a, b)$ represents the cosine similarity.

In Eq. (2), $\pi \otimes S$ indicates the instances correspond to the non-missing part in side information S, and U_{cos} is defined as the cosine utility function [46] by

$$U_{cos}(\pi', S) = \sum_{k=1}^{K} p_{k+} \|\langle \frac{p_{k1}^{(S)}}{p_{k+}}, \frac{p_{k2}^{(S)}}{p_{k+}}, \cdots, \frac{p_{kK}^{(S)}}{p_{k+}}\rangle\|, \qquad (4)$$

where $\pi' = \pi \otimes S$, $\langle \frac{p_{k1}^{(S)}}{p_{k+}}, \frac{p_{k2}^{(S)}}{p_{k+}}, \cdots, \frac{p_{kK}^{(S)}}{p_{k+}}\rangle$ indicates a row vector, $p_{kj}^{(S)} = n_{kj}^{(S)}/n\tau$ and $p_{k+} = n_{k+}/n\tau$, $1 \le j, k \le K$, and τ is the proportion of non-missing labels in S (*i.e.*, $S(x) = 1$ or 2 in Eq. (1)). Here, $p_{kj}^{(S)}$ and p_{k+} are defined based on the normalized contingency matrix to compute the co-occurrence of two discrete variables, where $n_{k+} = \sum_{j=1}^{K} n_{kj}^{(S)}$, and $n_{kj}^{(S)}$ represents the number of data instances (*i.e.*, superpixels) that simultaneously belongs to the cluster $\mathcal{C}_j^{(S)}$ in S and cluster $\mathcal{C}_k$ in π'. In Eq. (4), we use ℓ_2 norm to measure the distribution of the projection S to the k^{th} cluster in π', and linearly combine K distributions by utilizing their cluster size in π' as weights to obtain U_{cos}. We leverage U_{cos} to measure the similarity of two partitions, rather than two instances, with larger U_{cos} value indicating more similar partitions.

Taking a close look at Eq. (2), the benefits of the proposed MvSGC lie in several points. (1) We provide a unified clustering model that jointly considers feature similarity and side formation. By this means, we employ side information to guide the clustering process by feature similarity, which in turn, helps to recover the missing observations in the side information and thus improves the robustness of using unsupervised prior. (2) We compute the point-to-centroid feature distance with cosine similarity, which could capture non-spherical cluster structures and perform more efficiently than the squared Euclidean distance when dealing with high-dimensional and sparse features (*e.g.*, BoW features). (3) We introduce the cosine utility function U_{cos} to measure the similarity between the partition result π and the side information S. Owing to U_{cos}, our model finds the clustering solution that reveals the intrinsic structure from data itself and also agrees with the partition-level side information as much as possible. (4) We leverage multi-view weight learning to integrate multiple features, where the weight of each view is adaptively learned during the clustering process. By this adaptive fusion strategy, we employ a simple yet effective way to integrate the complementary information across multiple views, which improves robustness of MvSGC to handle the complex natural images.

D. Optimization Solution

The proposed objective function in Eq. (2) can be divided into two subproblems as constrained clustering and weight learning. Thus, we solve them in an alternative way, where we update one by keeping the other fixed and repeat the process iteratively until convergence. However, it is non-trivial to optimize Eq. (2) due to its element-wise formulation, where the augmented Lagrangian method cannot be directly used to

take the derivative of each variables. To address this challenge, we first focus on the second term in Eq. (2) and provide a new insight of the objective function.

1) Update π With Fixed w_i: U_{cos} in Eq. (2) measures the similarity at a partition-level. To unify it into a K-means-like form, the following lemma is introduced to measure the dissimilarity of two partitions with a distance function.

Lemma 1: Given two partitions H and S to separate a set of data instances X into K clusters, we have

$$\sum_{k=1}^{K}\sum_{s\in\mathcal{C}_k} f(s, m_k^{(S)}) \propto -U_{cos}(H, S), \tag{5}$$

where s represents the one-hot vector of one row in S, and $\mathcal{C}_1, \mathcal{C}_2, \cdots, \mathcal{C}_K$ are K clusters in H, $m_k^{(S)}$ is the k^{th} centroid vector of S according to $\mathcal{C}_k$ in H and f is the cosine distance.

Proof: Based on the point-to-centroid distance [47], we could rewrite the cosine distance as

$$f(a, b) = \phi(a) - \phi(b) - (a - b)^\top \nabla\phi(b), \tag{6}$$

where a, b are the non-zero vectors of the same size and $\phi(a) = \|a\|$. Then, we have

$$\begin{aligned}
&\sum_{k=1}^{K}\sum_{s\in\mathcal{C}_k} f(s, m_k^{(S)}) \\
&\quad= \sum_{k=1}^{K}\sum_{s\in\mathcal{C}_k} (\phi(s) - \phi(m_k^{(S)}) - (s - m_k^{(S)})^\top \nabla\phi(m_k^{(S)})) \\
&\quad= \alpha - \sum_{k=1}^{K} |\mathcal{C}_k| \phi(m_k^{(S)}) - \beta,
\end{aligned}$$

where $\alpha \equiv \sum_{k=1}^{K}\sum_{s\in\mathcal{C}_k}\phi(s)$ and $\beta \equiv \sum_{k=1}^{K}\sum_{s\in\mathcal{C}_k} (s - m_k^{(S)})^\top \nabla\phi(m_k^{(S)})$. As S is given in advance, α part is a constant and β part equals zeros due to the definition of centroid. Upon the variables in Eq. (4), it is straightforward to calculate the centroids as

$$m_k^{(S)} = \langle \frac{p_{k1}^{(S)}}{p_{k+}}, \frac{p_{k2}^{(S)}}{p_{k+}}, \cdots, \frac{p_{kK}^{(S)}}{p_{k+}} \rangle. \tag{7}$$

According to Eq. (4) and the deduction above, Eq. (5) holds and we finish the proof. □

Remark 1: In addition to employ utility function to calculate the similarity of two partitions, Lemma 1 gives a way to calculate the dissimilarity by a distance function. By this means, we have a new insight of the objective function in Eq. (2), which can be rewritten as the following:

$$\min_{\pi} \sum_{k=1}^{K}\sum_{x\in\mathcal{C}_k}\sum_{i=1}^{r} w_i^\gamma f(x^{(i)}, m_k^{(i)}) + \lambda \sum_{k=1}^{K}\sum_{s\in\mathcal{C}_k\cap S} f(s, m_k^{(S)}), \tag{8}$$

where $\mathcal{C}_k \cap S$ indicates the instances in $\mathcal{C}_k$ with non-missing side information.

In Eq. (8), each part has a K-means-like form. To achieve an efficient K-means solution, an auxiliary matrix B is first introduced. Specifically, we separate the data in each descriptor $X^{(i)}$ into two parts $X_1^{(i)}$ and $X_2^{(i)}$, where the instances in $X_1^{(i)}$ have non-missing side information in S and those in $X_2^{(i)}$ do not have. Then the auxiliary matrix B is organized as

$$B = \begin{bmatrix} X_1^{(1)} & X_1^{(2)} & \cdots & X_1^{(r)} & S' \\ X_2^{(1)} & X_2^{(2)} & \cdots & X_2^{(r)} & 0 \end{bmatrix}.$$

where S' is the non-missing part of S. $B = \{b\}$ consists of $r + 1$ parts by concatenating multi-view features and side information, *i.e.*, $b = \langle b^{(1)}, \cdots, b^{(r)}, b^{(S)} \rangle$; while for those instances without side information, zeros are used to fill up. Based on the auxiliary matrix B, we provide the following theorem to solve Eq. (8) in a neat mathematical way.

Theorem 1: Given multi-view data $X = [X^{(1)}, \cdots, X^{(r)}]$, side information S and an auxiliary matrix B, we have

$$\begin{aligned}
&\min_{\pi} \sum_{k=1}^{K}\sum_{x\in\mathcal{C}_k}\sum_{i=1}^{r} w_i^\gamma f(x^{(i)}, m_k^{(i)}) - \lambda U_{cos}(\pi \otimes S, S) \\
&\qquad\Leftrightarrow \min_{\pi} \sum_{k=1}^{K}\sum_{b\in\mathcal{C}_k} f'(b, m_k),
\end{aligned} \tag{9}$$

where b denotes one instance (or row) in B and $m_k = \langle m_k^{(1)}, m_k^{(2)}, \cdots, m_k^{(r)}, m_k^{(S)} \rangle$ is calculated as

$$m_k^{(i)} = \frac{\sum_{b\in\mathcal{C}_k} b^{(i)}}{|\mathcal{C}_k|}, m_k^{(S)} = \frac{\sum_{b\in\mathcal{C}_k\cap S} b^{(S)}}{|\mathcal{C}_k \cap S|}, \tag{10}$$

$1 \le i \le r$, and the distance function f' is computed by

$$f'(b, m_k) = \underbrace{\sum_{i=1}^{r} w_i^\gamma f(b^{(i)}, m_k^{(i)})}_{\text{multi-view features}} + \underbrace{\lambda I(b \in S) f(b^{(S)}, m_k^{(S)})}_{\text{partition-level side information}}, \tag{11}$$

where $I(b \in S) = 1$ means that S contains the side information for the instance b; and 0 otherwise.

Proof: It is easy to proof that

$$\begin{aligned}
&\sum_{k=1}^{K}\sum_{b\in\mathcal{C}_k} f'(b, m_k) \\
&\quad= \sum_{k=1}^{K}\sum_{b\in\mathcal{C}_k} (\sum_{i=1}^{r} w_i^\gamma f(b^{(i)}, m_k^{(i)}) + \lambda I(b \in S) f(b^{(S)}, m_k^{(S)})) \\
&\quad= \sum_{k=1}^{K}\sum_{x\in\mathcal{C}_k}\sum_{i=1}^{r} w_i^\gamma f(x^{(i)}, m_k^{(i)}) + \lambda \sum_{k=1}^{K}\sum_{s\in\mathcal{C}_k\cap S} f(s, m_k^{(S)}).
\end{aligned}$$

According to Lemma 1, Theorem 1 holds and we complete the proof. □

Based on Theorem 1 and the auxiliary matrix B, we finally transform Eq. (2) as a K-means-like problem in Eq. (9), which could be solved by the centroid update rules and distance function from Eq. (10) and Eq. (11).

2) Update w_i With Fixed π: By omitting the unrelated terms to w_i, Eq. (2) can be equivalently converted as

$$\min_{w_i\ge 0} \sum_{i=1}^{r} w_i^\gamma D_i \quad \text{s.t.} \sum_{i=1}^{r} w_i = 1, \tag{12}$$

Algorithm 1 MvSGC Clustering Algorithm

Input: Multi-view feature matrix $X = [X^{(1)}, \ldots, X^{(r)}]$, side information S, cluster number K, two parameters λ, γ.
Initial: $w_i = 1/r$.
1: Build the auxiliary matrix B;
2: **while** *not converged* **do**
3: Randomly select K instances as centroids;
4: **repeat**
5: Assign each instance to its closest centroid by the distance function in Eq. (11);
6: Update centroids by Eq. (10);
7: **until** the objective value in Eq. (2) remains unchanged.
8: Update w_i for each view by Eq. (15);
9: **end while**
Output: The final clustering result π.

where $D_i \equiv \sum_{k=1}^{K} \sum_{x \in \mathcal{C}_k} f(x^{(i)}, m_k^{(i)})$. To make Eq. (12) unconstrained, we write its Lagrange function as

$$\mathcal{L} = \sum_{i=1}^{r} w_i^{\gamma} D_i - \delta(\sum_{i=1}^{r} w_i - 1), \tag{13}$$

where δ is the Lagrange multiplier. By setting the derivative of $\mathcal{L}$ w.r.t w_i to zero, for $\forall i^{th}$ view, we have

$$w_i = (\frac{\delta}{\gamma D_i})^{\frac{1}{\gamma - 1}}. \tag{14}$$

To meet the KKT condition, we substitute Eq. (14) into the constraint $\sum_{i=1}^{r} w_i = 1$, and finally get the optimal weight for each view as following:

$$w_i = \frac{(\gamma D_i)^{\frac{1}{1-\gamma}}}{\sum_{i=1}^{r} (\gamma D_i)^{\frac{1}{1-\gamma}}}. \tag{15}$$

After π being given, D_i is only calculated by the i^{th} view's feature. Thus, our approach can learn the weight of each view adaptively, making the feature with more clear cluster structure has a larger weight. The parameter $\gamma > 1$ is employed to control the weight distribution among different views, where $\gamma \to 1$ leads to a sharp distribution, and $\gamma \to \infty$ makes equal weights. Since different features always have different performance and advantage on various cases, by using Eq. (15), we naturally leverage the complementary information across multiple views to cope with different image groups.

We alternatively update π and w_i until Eq. (2) becomes converged. The proposed MvSGC clustering algorithm is summarized by *Algorithm 1*. After having the partition π for all the superpixels, we may directly obtain the cosegmentation results by using cluster labels. However, due to our clustering result is on a superpixel-level, we further refine our cosegmentation results to a pixel-level by using the dense CRF framework [48], [49], as following many previous works [8], [10], [21].

E. Discussion

1) Convergence Analysis: An alternative optimization solution is given by Algorithm 1, which involves two steps:1) constrained clustering; and 2) multi-view weight learning. Although Step 1 is not the standard K-means, it has the convergence guarantee in both theoretical and practical perspectives. On the other side, Step 2 is a convex problem with respect to w_i, which has a closed-form solution given by Eq. (15). Therefore, by solving these two steps alternatively, Algorithm 1 converges to a local solution.

2) Time Complexity: According to Theorem 1, Step 1 can be solved by a K-means-like optimization with a modified distance function in Eq. (11) and centroid update rules in Eq. (7). Hence, it enjoys almost the same time complexity with standard K-means, $\mathcal{O}(T_1 K n(\sum_{i=1}^{r} d^{(i)} + K))$, where T_1 is the iteration number, K is the cluster number, n and $d^{(i)}$ are the instance number and feature number in $X^{(i)}$, respectively. It is worth noting that, after Step 1 being proceeded, D_i in Eq. (15) is fixed. Thus, the time cost for Step 2 could be omitted. Let T_2 be the iteration number of the outer loop in Algorithm 1, the final time complexity of our MvSGC algorithm is $\mathcal{O}(T_2(T_1 K n(\sum_{i=1}^{r} d^{(i)} + K)))$. Usually, we have $K \ll n$ and $d^{(i)} \ll n$, so the algorithm is roughly linear to the instance number, which indicates MvSGC is suitable for large-scale datasets.

IV. Experiment

A. Experimental Setting

1) Datasets and Criteria: We conduct the experiment for image cosegmentation on three benchmark RGB datasets, including iCoseg dataset[1] [1], MSRC dataset[2] and Internet dataset[3] [6]. Besides, one RGB-D image dataset[4] [21] is also used to test our approach. Two widely-used criteria are employed for the quantitative evaluation, which are precision, denoted as *Pre* (*i.e.*, the ratio of correctly labeled pixels of both foreground and background); and intersection over union, denoted as *IoU* (*i.e.*, the intersection over union of the result and the ground-truth segmentation). Both these two metrics are positive measures and ranged from 0 to 1.

2) Compared Methods: We compare our algorithm with 11 state-of-the-art methods, including (1) six graph-based methods such as Kim11 [17], Rub13 [6], Fu15 [21], Jer16 [38], Quan16 [8] and Ren18 [39]; (2) five clustering-based ones such as Jou12 [9], Lee15 [10], Liu15 [11], Sun16 [36] and Tao17 [14]. For Jou12, Lee15, Tao17 and Liu15, we run the author's code with recommended parameter setting. For the other methods, we directly use the results provided by the authors. Moreover, for a fair comparison, we run Liu15 [11] for cosegmentation by using the same side information and features to our method. Note that, since Liu15 can only work with single view, we test it under each view individually and post the best result.

3) Implementation Details: In this paper, we obtained the saliency prior by using the deep hierarchical saliency network (DHSNet) [50], which provides a fast and end-to-end way to obtain saliency map from image. For the multi-view features, we used three BoW histograms and one deep CNN feature, computed by SIFT [51], Texton [52], LAB colors [53], and the CNN model [54], respectively. In details, for each BoW descriptor, we obtained 300 words by performing K-means clustering on each image group with

[1] http://chenlab.ece.cornell.edu/projects/touch-coseg/
[2] http://research.microsoft.com/en-us/projects/objectclassrecognition/
[3] http://people.csail.mit.edu/mrub/ObjectDiscovery/
[4] http://hzfu.github.io/proj_rgbdseg.html

TABLE I

COMPARISON RESULTS OF SEGMENTATION PERFORMANCE (%) BETWEEN MVSGC AND OTHER METHODS ON THE ICOSEG DATASET OF 31 IMAGE GROUPS, THE ICOSEG DATASET OF 38 IMAGE GROUPS, THE MSRC DATASET, AND THE THREE IMAGE GROUPS OF THE INTERNET DATASET. (RED BOLD FONT FOR THE BEST PERFORMANCE; BLUE ITALIC FOR THE SECOND)

Methods	iCoseg-31		iCoseg-38		MSRC		Internet-Airplane		Internet-Car		Internet-Horse	
	Pre	*IoU*	*Pre*	*IoU*	*Pre*	*IoU*	*Pre*	*IoU*	*Pre*	*IoU*	*Pre*	*IoU*
Kim11 [16]	77.9	41.5	-	-	57.7	35.5	80.2	7.9	68.9	0.04	75.1	6.4
Jou12 [9]	72.4	46.8	70.5	41.6	73.6	50.7	47.5	11.7	59.2	35.2	64.2	29.5
Rub13 [6]	89.3	68.4	-	-	87.7	68.1	88.0	55.8	85.4	64.4	82.8	51.7
Liu15 [11]	81.1	49.8	80.5	45.9	80.7	55.4	87.3	49.4	85.9	63.4	88.9	56.2
Fu15 [20]	88.1	60.2	89.0	58.3	-	-	-	-	-	-	-	-
Lee15 [10]	90.6	70.0	90.6	66.7	84.0	59.4	52.8	36.3	64.7	42.3	70.1	39.0
Jer16† [37]	-	71.7	91.9	72.0	*88.7*	71.0	-	-	-	-	-	-
Sun16† [35]	-	-	-	-	77.5	54.8	88.6	36.3	87.0	*73.4*	87.6	54.7
Quan16 [8]	*93.4*	**77.1**	*93.3*	**76.0**	-	-	*91.0*	*56.3*	*88.5*	66.8	*89.3*	*58.1*
Tao17 [13]	90.8	70.4	90.5	67.3	86.4	64.5	79.8	42.8	84.8	66.4	85.7	55.3
Ren18† [38]	-	73.5	-	-	-	*72.0*	88.3	47.7	83.5	62.4	83.2	49.4
MvSGC	**93.5**	*76.7*	**93.9**	*75.8*	**89.9**	**72.5**	**92.4**	**62.7**	**91.9**	**77.6**	**90.1**	**61.9**

† indicates the results are reported by the authors' paper.

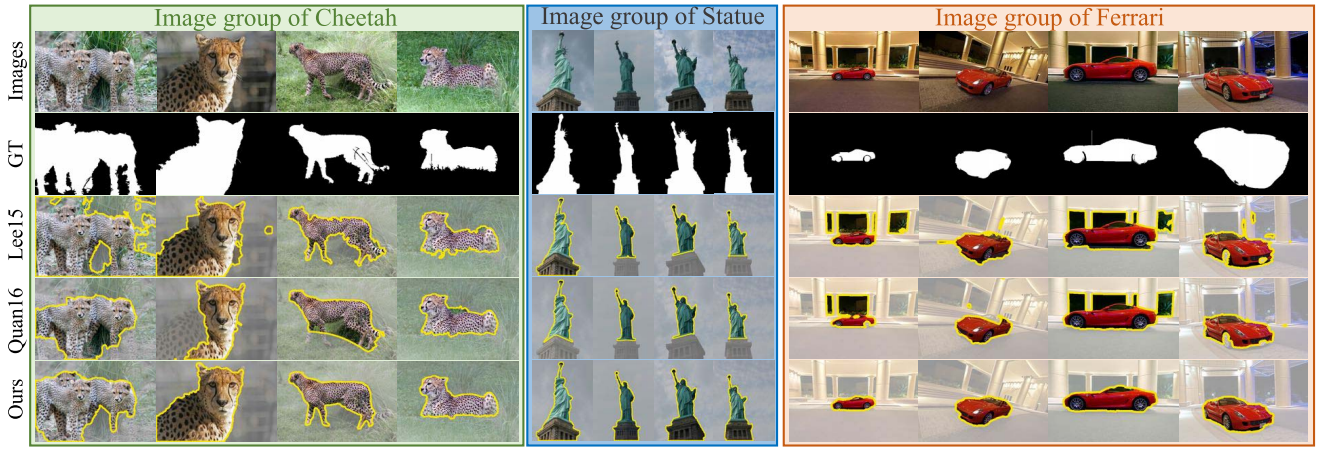


Fig. 3. Visual comparison results between Lee15 [10], Quan16 [8] and our MvSGC on iCoseg (Best viewed in color).

the superpixel-level features. Thus, each BoW was represented as a 300-D feature vector. For the CNN features, we used the last convolutional layer of the CNN model [54] pre-trained on ImageNet, to obtain 512 feature maps (17×17) for each image, and then upsampled each feature map to the original size of the image. We finally generated a 512-D feature vector by performing max pooling operation among these maps on the superpixel-level. In our experiments, we set $\lambda = 1000$ and $\gamma = 100$ as the default setting. All the experiments were conducted by MATLAB on a 64-bit Windows platform with two Intel Core i7 3.4GHz CPUs, one Titan-X GPU and 32GB RAM.

B. Image Cosegmentation

1) iCoseg Dataset: The iCoseg dataset is a widely-used benchmark for image cosegmentation, consisting of 38 image groups with 643 images in total. We fist test our approach by following the same setting in [6], which selects 31 image groups with 530 images. Table I shows the cosegmentation performance of our proposed MvSGC and other methods on the subset (31 image groups) of iCoseg dataset, where we achieve comparable results to one of the state-of-the-art graph-based models, *i.e.*, Quan16 [8]. This fully validates the effectiveness of our clustering model for the image cosegmentation task. However, MvSGC performs slightly worse than Quan16 [8] by *IoU*, which might be due to the reasons that *IoU* has a bias towards small objects and our saliency prior may overlook the small common objects. On the other hand, we outperform all the clustering-based methods and the other graph-based methods, with a clear improvement. One may note that, we improve our previous work, Tao17 [14], with 2.7% *Pre* and 6.3% *IoU*, which demonstrates this paper is a substantial extension to [14]. We also test MvSGC and the other methods on the whole iCoseg dataset (38 image groups), where the similar conclusion could be made.

Fig. 3 provides a visual comparison of selected 3 image groups between Quan16, Lee15 and our approach, where Quan16 and Lee15 are the top performers among graph-based and clustering-based compared methods, respectively. As can be seen, Quan16 and Lee15 suffer from the incomplete object segmentation and background noises. For the example of *Ferrari* image group, they both miss the tire of the car and wrongly segment the black color background as foreground. In contrast, benefiting from multi-view feature integration and the guidance of our side information, we obtain more accurate object segmentation within each image, and suppress the common background regions among images.

2) MSRC Dataset: The MSRC dataset consists of 14 image groups with totally 410 images, where each image group

contains similar foreground objects. As shown in Table I, the proposed MvSGC outperforms all the other methods on this dataset. Compared with clustering-based methods, the great improvements (*e.g.*, round 6% *Pre* and 13% *IoU* higher than Lee15) indicates our MvSGC model as a very promising clustering method for image cosegmentation. Moreover, our approach improves nearly 9% *Pre* and 17% *IoU* compared with Liu15 [11], which demonstrates the significant superiority of using cosine similarity to the squared Euclidean distance.

3) Internet Dataset: The Internet dataset is one of the most challenging datasets for image cosegmentation task. It collects thousands of images from the Internet according to three categories of Airplane, Car and Horse, where each image group has some noisy images. By following [6], we use a subset of the Internet dataset as 100 images per class. Table I summarizes the performance of our approach and other compared methods on three image groups of this dataset, where we achieve the best *Pre* and *IoU* on all the cases. For the Internet dataset, objects in each image group usually share with the same semantic information (*i.e.*, image class), yet with quite different appearances. Thus, the methods without considering higher-level concepts may lose their effectiveness. In contrast, by utilizing the deep CNN representation, Quan16 [8] and our MvSGC outperform other methods significantly, which demonstrates the effectiveness of using higher-level features for image cosegmentation. In another hand, compared with Quan16, we clearly improve the performance by round 6%, 11%, and 3% *IoU* on these three classes.

To sum up, the proposed MvSGC performs better than several state-of-the-art cosegmentation methods in most cases of Table I, indicating the robustness of our approach when dealing with different image groups. This is mainly due to the unified clustering model provided by our method, which jointly considers the instance-level and partition-level similarity, and the benefits from our multi-view weight learning.

C. Model Discussion

1) Time Efficiency: We compare our proposed MvSGC with two clustering-based cosegmentation methods, Jou12 [9] and Lee15 [10], in terms of the running time. It is worth noting that, though Jou12 and Lee15 formulate cosegmentation as a clustering problem, they both employ graph-based clustering algorithms, *e.g.*, the spectral clustering term in [9] and the random walk process in [10], which cost much on the graph construction process and may suffer from a high time complexity (*e.g.*, the eigenvalue decomposition in spectral clustering). In contrast, we solve our clustering problem with a neat K-means-like optimization, and thus have a roughly linear time complexity. As shown in Table II, our approach is over 4 times faster than Jou12 and 3 times faster than Lee15 on the iCoseg dataset, while similar comparison result appears on MSRC. Moreover, MvSGC is round 19 times faster than Jou12 and 9 times faster than Lee15 on the Internet dataset. This is mainly because the image group in Internet has more images than the other datasets. We also show the execution time of our clustering process. As can be seen, our clustering algorithm costs quite little, which could be omitted compared with the total running time. Actually, feature engineering is the most time-consuming part in our model. To sum up, the proposed MvSGC is a highly efficient clustering algorithm, which exhibits huge potential to the large-scale problem.

TABLE II

COMPARISON OF COMPUTATIONAL TIME BETWEEN THE PROPOSED MVSGC AND TWO CLUSTERING-BASED METHODS

Dataset	#Image	Jou12 (*hour*)	Lee15 (*hour*)	MvSGC	
				Clustering (*sec.*)	Total (*hour*)
iCoseg	643	8.25	5.83	**193.61**	1.99
MSRC	410	4.02	1.97	**39.66**	0.49
Internet	300	8.26	3.72	**57.49**	0.43

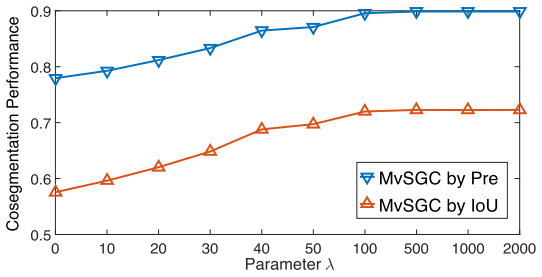


Fig. 4. Parameter analysis for MvSGC on the MSRC dataset.

TABLE III

COMPARISON OF MULTI-VIEW FEATURES AND THE EUCLIDEAN DISTANCE BASED CONSTRAINED CLUSTERING LIU15 [11] BY *IoU*% (RED BOLD FONT FOR THE BEST PERFORMANCE; BLUE ITALIC FOR THE SECOND)

Datasets	Liu15	SIFT	Texton	LAB	CNN	MvSGC
iCoseg	45.9	39.1	46.3	*57.0*	44.7	**75.8**
MSRC	55.4	41.9	38.1	50.5	*55.6*	**72.5**
Internet	*63.9*	25.2	32.6	30.9	41.7	**77.0**

2) Parameter Analysis: There are two parameters, *i.e.*, γ and λ, used in our model. We employ γ to control the weight distribution among multiple views. Generally, we may achieve better performance through tuning this parameter. However, due to our unsupervised setting, we empirically set $\gamma = 100$ in all the experiments, as suggested by [45]. The parameter λ in Eq. (2) is the key factor that balances two kinds of similarities in our model, which are the instance-level similarity computed by multi-view features with cosine similarity, and the partition-level similarity given by side information with cosine utility function. We expect to set λ a relatively large value, to make the partition-level similarity comparable to the magnitude of multi-view features. Here, to explore the impact of λ to our model, we vary λ from the set $\{0, 10, 20, 30, 40, 50, 100, 500, 1000, 2000\}$, and test MvSGC on the MSRC dataset with different parameters. As shown in Fig. 4, our performance generally goes up with the increase of λ and keeps stable when $\lambda > 100$. This indicates our approach is insensitive to the parameter λ by setting it in a certain range.

3) Multi-View Features: Table III gives the performance of different views in our model. In details, we run our method on each single view without using saliency

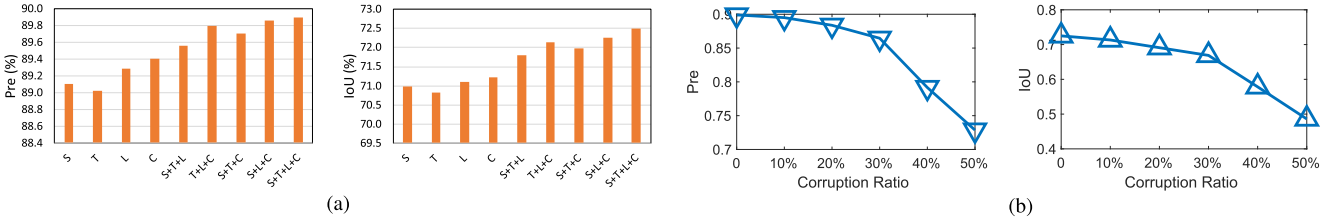


Fig. 5. Impact of feature combinations and noisy priors to our proposed MvSGC model on the MSRC image dataset. (a) Cosegmentation performance of MvSGC with different feature combinations, where S, T, L, and C indicate SIFT [51], Texton [52], LAB [53] and CNN [54] features, respectively. (b) MvSGC with noisy priors by setting different corruption ratios.

prior (*i.e.*, $\lambda = 0$). As can be seen, the LAB feature outperforms best on iCoseg, while CNN features performs much better than the others on MSRC and Internet. This is mainly due to various properties of different image datasets. Specifically, most image groups in the iCoseg dataset share with consistent appearance features with little illumination change [9], leading to the superiority of LAB color feature on this dataset. In another hand, images in the MSRC and Internet dataset usually have semantically similar foreground objects with large appearance variations. Hence, deep CNN features containing higher-level information (*i.e.*, image class) achieve the best in these two datasets. Different advantages of different features motivate us to improve the robustness of our clustering process by integrating these features with multi-view weight learning.

Table III also shows the cosegmentation performance of Liu15 [11] on three datasets. Different from our method, Liu15 computes feature similarity by the squared Euclidean distance and can only work with individual feature descriptor. In the experiment, we feed Liu15 with the same side information to our MvSGC, and report its result of the best single view. One may note that, even without the guidance of side information, our best single-view result still outperforms Liu15 on the iCoseg dataset, and achieves similar performance to Liu15 on the MSRC. This fully demonstrates the benefit of using cosine similarity to the squared Euclidean distance. In summary, we employ multi-view features to handle various datasets, and utilize cosine similarity to better capture the cluster structure in feature space.

To further explore the impact of multi-view features to our approach, we conduct the proposed MvSGC model with different feature combinations, including four single-view features (*i.e.*, SIFT [51], Texton [52], LAB [53] and CNN [54] denoted by S, T, L and C, respectively) and four combinations of three-view features (*i.e.*, S+T+L, T+L+C, S+T+C, and S+L+C). As shown in Fig. 5(a), two observations could be made: 1) cosegmentation results with multiple features outperform the ones using individual feature; 2) multi-view features (T+L+C, S+T+C, and S+L+C) containing higher-level information improve the performance of MvSGC with only low-level features (S+T+L). This clearly shows the benefit of using multiple visual cues for image cosegmentation, as well as utilizing the abstract concepts in image classification given by the deep CNN features.

4) Saliency Priors: Fig. 6 shows the performance of our MvSGC under two different saliency priors, where AVG

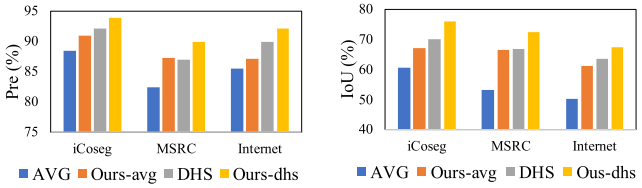


Fig. 6. Exploration of saliency prior to the proposed MvSGC on three datasets, where we test our algorithm by using the aggregated saliency prior given by [14] (denoted as AVG), and the saliency prior obtained by [50] (denoted as DHS). Ours-avg and Ours-dhs represent the results of performing MvSGC with prior AVG and DHS, respectively.

represents the aggregated saliency prior provided by [14], which is obtained from averaging the results of three saliency detection methods [55]–[57]; and DHS denotes the default saliency prior in our model given by DHSNet [50]. By feeding the same multi-view features, we perform MvSGC with AVG and DHS on three datasets, respectively. We also test these two priors by thresholding them as binary segmentation with the adaptive threshold [44]. As can be seen, our method consistently boosts the performance of saliency prior on different scenarios, which demonstrates that our method can effectively utilize the information from multi-view features to recover the missing observations in different saliency priors. Moreover, by giving a more powerful prior (*i.e.*, more credible "labels"), we improve the cosegmentation performance significantly. This indicates the effectiveness of MvSGC as a constrained clustering method.

Saliency prior may suffer from incorrect detection results and provide noisy labels to our clustering model. To alleviate this problem, we adopt a partial observation strategy to obtain the partition-level side information (see Section III-B), and recover the missing observations by using the correspondence of common objects across images. To explore the impact of severely noisy saliency priors, we randomly set the partition labels (induced by DHSNet [50]) as zeros according to a corruption ratio p. Fig. 5(b) shows the cosegmentation performance of MvSGC on the MSRC dataset by ranging p from 0 to 50% with a step size of 10%. As can be seen, our approach still achieves a good performance when $p \leq 30\%$, which shows the robustness of MvSGC to noisy priors.

D. RGB-D Image Cosegmentation

With the increasing of affordable RGB-D sensors, depth information gets more popular for image segmentation [58], which is useful to distinguish foreground object from cluttered

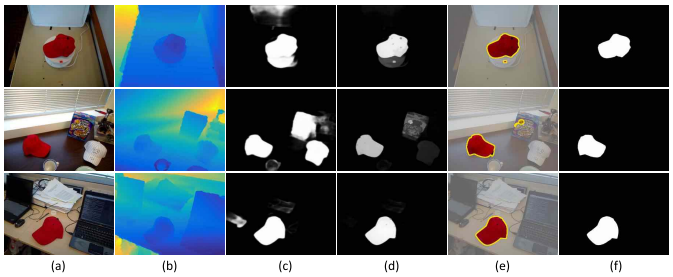


Fig. 7. Illustration of saliency prior incorporated with depth information, where (a) Images, (b) raw depth maps, (c) saliency prior *w/o* depth, (d) saliency prior + depth, (e) Our cosegmentation results and (f) Ground-truth annotations.

TABLE IV
COMPARISON RESULTS BETWEEN MVSGC AND OTHER METHODS ON THE RGB-D IMAGE DATASET (RED BOLD FONT FOR THE BEST PERFORMANCE; BLUE ITALIC FOR THE SECOND)

	Methods	*Pre* (%)	*IoU* (%)
w/o depth	Jou10 [5]	55.8	14.5
	Kim11 [16]	85.7	32.3
	Lee15 [10]	73.4	26.2
	MvSGC (w/o depth)	92.2	45.8
+ depth	Liu15 [11]	67.7	31.5
	Fu15 [20]	*93.3*	*47.9*
	Tao17 [13]	91.0	41.4
	MvSGC (+ depth)	**94.0**	**50.8**

background and preserve object boundaries. In light of this, Fu *et al.* [21] first extended RGB-D image segmentation to the multiple case, and employed the saliency priors incorporated with depth cues to highlight common objects among image group. Following this work, we apply our MvSGC clustering method to the RGB-D image cosegmentation task.

We conduct experiments on the RGB-D image dataset provided by [21], which contains 16 image groups (totally 193 images) with corresponding depth maps. Fig. 7 gives some examples of the *red hat* image group in this dataset. As shown in Fig. 7(c), the saliency prior given by [50], which is without depth information (denoted as *w/o* depth), may wrongly detect objects from the complex background. To alleviate this problem, we employ a multiple fusion strategy to incorporate depth cues into our side information, that is, to multiple our original saliency prior with the RGB-D based saliency maps provided by [21]. Fig. 7(d) shows the fusion result. As can be seen, the depth information weakens the saliency value of background regions effectively.

Table IV summarizes the performance of our MvSGC and other methods on the RGB-D image dataset. Overall, the methods containing depth information generally outperform the methods without using depth cues. As can be seen, MvSGC (+ depth) improves round 2% *Pre* and 5% *IoU* over MvSGC (*w/o* depth), which shows the effectiveness of adding depth prior into our side information. Moreover, compared with Fu15 [21], which is specifically designed for the RGB-D image data, our method still exceeds by round 1% *Pre* and 3% *IoU*. This shows the proposed MvSGC as a flexible clustering method to incorporate with different visual cues.

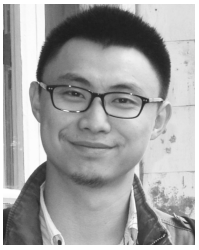
V. CONCLUSION

A novel Multi-view Saliency-Guided Clustering (MvSGC) algorithm was proposed with cosine similarity in this paper, which derives unsupervised saliency prior as a partition-level side information to guide the clustering process. The robustness of our method inherits from two aspects, *i.e.*, a unified clustering model that jointly considers the feature and partition similarity, and a multi-view weight learning approach that integrates various visual cues. By giving a new insight to the cosine utility function, we finally provided a K-means-like optimizing solution with theoretical guarantee, leading to the roughly linear time complexity of MvSGC. Experiments on three image datasets were conducted to demonstrate the effectiveness of our approach compared with state-of-the-art methods. Moreover, experiments on RGB-D images showcased the flexibility of our algorithm. In the future, we will explore multi-modality features for image cosegmentation, and apply our algorithm in the video object segmentation task.

REFERENCES

[1] 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, Jul. 2011.
[2] S. Vicente, C. Rother, and V. Kolmogorov, "Object cosegmentation," in *Proc. CVPR*, Jun. 2011, pp. 2217–2224.
[3] M. Rubinstein, C. Liu, and W. T. Freeman, "Joint inference in weakly-annotated image datasets via dense correspondence," *Int. J. Comput. Vis.*, vol. 119, no. 1, pp. 23–45, 2016.
[4] C. Rother, T. Minka, A. Blake, and V. Kolmogorov, "Cosegmentation of image pairs by histogram matching—Incorporating a global constraint into MRFs," in *Proc. CVPR*, Jun. 2006, pp. 993–1000.
[5] A. Joulin, F. R. Bach, and J. Ponce, "Discriminative clustering for image co-segmentation," in *Proc. CVPR*, Jun. 2010, pp. 1943–1950.
[6] M. Rubinstein, A. Joulin, J. Kopf, and C. Liu, "Unsupervised joint object discovery and segmentation in internet images," in *Proc. CVPR*, Jun. 2013, pp. 1939–1946.
[7] D. S. Hochbaum and V. Singh, "An efficient algorithm for Co-segmentation," in *Proc. ICCV*, Oct. 2009, pp. 269–276.
[8] R. Quan, J. Han, D. Zhang, and F. Nie, "Object co-segmentation via graph optimized-flexible manifold ranking," in *Proc. CVPR*, Jun. 2016, pp. 687–695.
[9] A. Joulin, F. R. Bach, and J. Ponce, "Multi-class cosegmentation," in *Proc. CVPR*, Jun. 2012, pp. 542–549.
[10] C. Lee, W.-D. Jang, J.-Y. Sim, and C.-S. Kim, "Multiple random walkers and their application to image cosegmentation," in *Proc. CVPR*, Jun. 2015, pp. 3837–3845.
[11] H. Liu and Y. Fu, "Clustering with partition level side information," in *Proc. ICDM*, Nov. 2015, pp. 877–882.
[12] H. Liu, Z. Tao, and Y. Fu, "Partition level constrained clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 10, pp. 2469–2483, Oct. 2017.
[13] B. Mirkin, "Reinterpreting the category utility function," *Mach. Learn.*, vol. 45, no. 2, pp. 219–228, 2001.
[14] Z. Tao, H. Liu, H. Fu, and Y. Fu, "Image cosegmentation via saliency-guided constrained clustering with cosine similarity," in *Proc. AAAI*, 2017, pp. 4285–4291.
[15] L. Mukherjee, V. Singh, and C. R. Dyer, "Half-integrality based algorithms for cosegmentation of images," in *Proc. CVPR*, Jun. 2009, pp. 2028–2035.
[16] L. Mukherjee, V. Singh, and J. Peng, "Scale invariant cosegmentation for image groups," in *Proc. CVPR*, Jun. 2011, pp. 1881–1888.
[17] G. Kim, E. P. Xing, L. Fei-Fei, and T. Kanade, "Distributed cosegmentation via submodular optimization on anisotropic diffusion," in *Proc. ICCV*, Nov. 2011, pp. 169–176.
[18] J. C. Rubio, J. Serrat, A. M. López, and N. Paragios, "Unsupervised co-segmentation through region matching," in *Proc. CVPR*, Jun. 2012, pp. 749–756.
[19] A. Faktor and M. Irani, "Co-segmentation by composition," in *Proc. ICCV*, Dec. 2013, pp. 1297–1304.

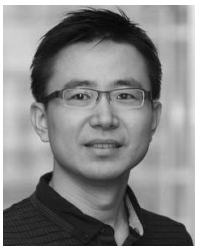
[20] F. Wang, Q. Huang, and L. J. Guibas, "Image co-segmentation via consistent functional maps," in *Proc. ICCV*, Dec. 2013, pp. 849–856.
[21] H. Fu, D. Xu, S. Lin, and J. Liu, "Object-based RGBD image co-segmentation with mutex constraint," in *Proc. CVPR*, Jun. 2015, pp. 4428–2236.
[22] C. Rother, 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.
[23] X. Dong, J. Shen, L. Shao, and M.-H. Yang, "Interactive cosegmentation using global and local energy optimization," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3966–3977, Nov. 2015.
[24] W. Wang and J. Shen, "Higher-order image co-segmentation," *IEEE Trans. Multimedia*, vol. 18, no. 6, pp. 1011–1021, Jun. 2016.
[25] G. Kim and E. P. Xing, "On multiple foreground cosegmentation," in *Proc. CVPR*, Jun. 2012. pp. 837–844.
[26] J. Ma, S. Li, H. Qin, and A. Hao, "Unsupervised multi-class co-segmentation via joint-cut over L_1 -manifold hyper-graph of discriminative image regions," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1216–1230, Mar. 2017.
[27] H. Fu, D. Xu, B. Zhang, S. Lin, and R. K. Ward, "Object-based multiple foreground video co-segmentation via multi-state selection graph," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3415–3424, Nov. 2015.
[28] 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.
[29] L. Wang, G. Hua, R. Sukthankar, Z. Niu, J. Xue, and N. Zheng, "Video object discovery and co-segmentation with extremely weak supervision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 10, pp. 2074–2088, Oct. 2017.
[30] Y. W. Zehuan Yuan and T. Lu, "Deep-dense conditional random fields for object co-segmentation," in *Proc. IJCAI*, 2017, pp. 3371–3377.
[31] F. Meng, H. Li, Q. Wu, B. Luo, and K. N. Ngan, "Weakly supervised part proposal segmentation from multiple images," *IEEE Trans. Image Process.*, vol. 26, no. 8, pp. 4019–4031, Aug. 2017.
[32] E. Kim, H. Li, and X. Huang, "A hierarchical image clustering cosegmentation framework," in *Proc. CVPR*, Jun. 2012, pp. 686–693.
[33] M. D. Collins, J. Xu, L. Grady, and V. Singh, "Random walks based multi-image segmentation: Quasiconvexity results and GPU-based solutions," in *Proc. CVPR*, Jun. 2012, pp. 1656–1663.
[34] W. Tao, K. Li, and K. Sun, "SaCoseg: Object cosegmentation by shape conformability," *IEEE Trans. Image Process.*, vol. 24, no. 3, pp. 943–955, Mar. 2015.
[35] F. Meng *et al.*, "Constrained directed graph clustering and segmentation propagation for multiple foregrounds cosegmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 11, pp. 1735–1748, Nov. 2015.
[36] J. Sun and J. Ponce, "Learning dictionary of discriminative part detectors for image categorization and cosegmentation," *Int. J. Comput. Vis.*, vol. 120, no. 2, pp. 111–133, 2016.
[37] K. Chang, T. L. Liu, and S. H. Lai, "From co-saliency to co-segmentation: An efficient and fully unsupervised energy minimization model," in *Proc. CVPR*, Jun. 2011, pp. 2129–2136.
[38] K. R. Jerripothula, J. Cai, and J. Yuan, "Image co-segmentation via saliency co-fusion," *IEEE Trans. Multimedia*, vol. 18, no. 9, pp. 1896–1909, Sep. 2016.
[39] Y. Ren, L. Jiao, S. Yang, and S. Wang, "Mutual learning between saliency and similarity: Image cosegmentation via tree structured sparsity and tree graph matching," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4690–4704, Sep. 2018.
[40] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "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.
[41] J. Shen, X. Hao, Z. Liang, Y. Liu, W. Wang, and L. Shao, "Real-time superpixel segmentation by DBSCAN clustering algorithm," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5933–5942, Dec. 2016.
[42] 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.
[43] X. Cao, Z. Tao, B. Zhang, H. Fu, and W. Feng, "Self-adaptively weighted co-saliency detection via rank constraint," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 4175–4186, Sep. 2014.
[44] Y. Jia and M. Han, "Category-independent object-level saliency detection," in *Proc. CVPR*, Dec. 2013, pp. 1761–1768.
[45] X. Cai, F. Nie, and H. Huang, "Multi-view k-means clustering on big data," in *Proc. IJCAI*, 2013, pp. 2598–2604.
[46] J. Wu, H. Liu, H. Xiong, J. Cao, and J. Chen, "K-means-based consensus clustering: A unified view," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 1, pp. 155–169, Jan. 2015.
[47] J. Wu, H. Xiong, C. Liu, and J. Chen, "A generalization of distance functions for fuzzy$c$-means clustering with centroids of arithmetic means," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 3, pp. 557–571, Jun. 2012.
[48] P. Kráhenbühl and V. Koltun, "Efficient inference in fully connected CRFS with Gaussian edge potentials," in *Proc. NIPS*, 2011, pp. 109–117.
[49] P. Krähenbühl and V. Koltun, "Parameter learning and convergent inference for dense random fields," in *Proc. ICML*, Feb. 2013, pp. 513–521.
[50] N. Liu and J. Han, "DHSNet: Deep hierarchical saliency network for salient object detection," in *Proc. CVPR*, Jun. 2016, pp. 678–686.
[51] C. Liu, J. Yuen, and A. Torralba, "SIFT flow: Dense correspondence across scenes and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 978–994, May 2011.
[52] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. ICCV*, Oct. 2003, pp. 1470–1477.
[53] T. Deselaers and V. Ferrari, "Global and efficient self-similarity for object classification and detection," in *Proc. CVPR*, Jun. 2010, pp. 1633–1640.
[54] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *Proc. BMVC*, Aug. 2014. pp. 125–136.
[55] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. CVPR*, Jun. 2013, pp. 3166–3173.
[56] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. CVPR*, Jun. 2013, pp. 1155–1162.
[57] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata," in *Proc. CVPR*, Jun. 2015, pp. 110–119.
[58] S. Gupta, R. B. Girshick, P. A. Arbeläez, and J. Malik, "Learning rich features from RGB-D images for object detection and segmentation," in *Proc. ECCV*, Sep. 2014, pp. 345–360.



Zhiqiang Tao received the B.E. degree in software engineering from the School of Computer Software, and the M.S. degree in computer science from the School of Computer Science and Technology, Tianjin University, Tianjin, China, in 2012 and 2015, respectively. He is currently pursuing the Ph.D. degree with Northeastern University, Boston. His research interests include data cluster analysis, subspace learning, ensemble clustering, and unsupervised deep representation learning.

Hongfu Liu received the bachelor's and master's degrees in management information systems from the School of Economics and Management, Beihang University, in 2011 and 2014, respectively, and the Ph.D. degree in computer engineering from Northeastern University, Boston, MA, USA, in 2018. He is currently a tenure-track Assistant Professor with the Michtom School of Computer Science, Brandeis University. His research interests include data mining and machine learning, with special interests in ensemble learning. He served as a reviewer for many IEEE TRANSACTIONS journals, including TKDE, TNNLS, TIP, and TBD. He also served on the Program Committee for the conferences, including AAAI, IJCAI, and NIPS. He is an Associate Editor of the *IEEE Computational Intelligence Magazine*.

Huazhu Fu (SM'18) received the Ph.D. degree in computer science from Tianjin University, China, in 2013. He was a Research Fellow with Nanyang Technological University, Singapore, for two years. From 2015 to 2018, he was a Research Scientist with the Institute for Infocomm Research, Agency for Science, Technology and Research, Singapore. He is currently a Senior Scientist with the Inception Institute of Artificial Intelligence, Abu Dhabi, United Arab Emirates. His research interests include computer vision, image processing, and medical image analysis. He is an Associate Editor of IEEE ACCESS and *BMC Medical Imaging*.



Yun Fu (S'07–M'08–SM'11–F'19) received the B.Eng. degree in information engineering and the M.Eng. degree in pattern recognition and intelligence systems from Xian Jiaotong University, China, and the M.S. degree in statistics and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign. He has been an Interdisciplinary Faculty Member with the College of Engineering and the Khoury College of Computer and Information Sciences, Northeastern University, since 2012. He has extensive publications in leading journals, books/book chapters, and international conferences/workshops. His research interests include machine learning, computational intelligence, big data mining, computer vision, pattern recognition, and cyber-physical systems. He is a fellow of IAPR, OSA, and SPIE, a Lifetime Distinguished Member of ACM, a Lifetime Member of AAAI and Institute of Mathematical Statistics, a member of ACM Future of Computing Academy, Global Young Academy, AAAS, INNS, and Beckman Graduate Fellow from 2007 to 2008. He received seven Prestigious Young Investigator Awards from NAE, ONR, ARO, the IEEE, INNS, UIUC, and Grainger Foundation, nine Best Paper Awards from the IEEE, IAPR, SPIE, and SIAM, and many major Industrial Research Awards from Google, Samsung, and Adobe. He is currently an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEANING SYSTEMS (TNNLS) and the IEEE TRANSACTIONS OF IMAGE PROCESSING (TIP). He serves as an associate editor, a chair, a PC member, and a reviewer for many top journals and international conferences/workshops.