A New Method and Benchmark for Detecting Co-Saliency Within a Single Image

Hongkai Yu, *Member, IEEE*, Kang Zheng, Jianwu Fang, Hao Guo, and Song Wang, *Senior Member, IEEE*

***Abstract*—Recently, saliency detection in a single image and co-saliency detection in multiple images have drawn extensive research interest in the vision and multimedia communities. In this paper, we investigate a new problem of co-saliency detection within a single image, i.e., detecting within-image co-saliency. By identifying common saliency within an image, e.g., highlighting multiple occurrences of an object class with similar appearance, this work can benefit many important applications, such as the detection of objects of interest, more robust object recognition, reduction of information redundancy, and animation synthesis. We propose a new bottom-up method to address this problem. Specifically, a large number of object proposals are first detected from the image. Then we develop an optimization algorithm to derive a set of proposal groups, each of which contains multiple proposals showing good common saliency in the image. For each proposal group, we calculate a co-saliency map and then use a low-rank based algorithm to fuse the maps calculated from all the proposal groups for the final co-saliency map in the image. In the experiment, we collect a new benchmark dataset of 664 color images (two subsets) for within-image co-saliency detection. Experiment results show that the proposed method can better detect the within-image co-saliency than existing algorithms. The experimental results also show that the proposed method can be applied to detect the repetitive patterns in a single image and detect the co-saliency in multiple images.**

***Index Terms*—Within-image co-saliency, convex optimization, low-rank based fusion.**

I. Introduction

LARGE scale multimedia data is generated everyday due to the fast technology development (such as image, video, audio, text) from different multimedia sensors and devices. Human beings are always expecting to efficiently and accurately extract salient and meaningful information from the large scale multimedia data. This paper focuses on the saliency detection from image data, which has a wide range of multimedia applications, such as fast image retrieval from database [2], face video application and encoding [3], object recognition [4], segmentation [5], etc. Image-based saliency detection, which is to highlight the salient/interested regions in images, has drawn extensive interest in the vision and multimedia communities in the past decade.

Fig. 1. Illustrations of different saliency detection problems. (a) Within-image saliency detection. (b) Cross-image co-saliency detection, where co-saliency is detected across three images. (c) The proposed within-image co-saliency detection. First row: images. Second row: ground-truth saliency/co-saliency maps.

Research in this area started with saliency detection in a single image, i.e., *within-image saliency* detection, which aims at highlighting the visually standing-out regions/objects/structures from the surrounding background [6]–[10], as illustrated in Fig. 1(a). More recently, co-saliency detection in multiple images, e.g., *cross-image co-saliency* detection [4], [11]–[14], has been attracting much attention with many successful applications [5], [15]–[17]. As illustrated in Fig. 1(b), cross-image co-saliency detection aims to detect the common saliency, e.g., red-clothed soccer players, that are present in all three images. In this paper, we investigate a new problem of detecting co-saliency within a single image, i.e., *within-image co-saliency* detection, which aims to highlight the common saliency within an image. An example is shown in Fig. 1(c), where the two red-clothed players show good within-image co-saliency, but the white-clothed player does not because only one white-clothed player is present in the image.

Within-image co-saliency detection might benefit many important applications in computer vision and multimedia. For example, it can be used to help detect multiple instances of

Manuscript received March 15, 2019; revised October 14, 2019; accepted January 24, 2020. Date of publication February 6, 2020; date of current version November 18, 2020. This work was supported in part by the National Natural Science Foundation of China under Grants NSFC-U1803264, NSFC-61671325, 61672376, and 61603057. This article was presented in part at the 32nd Conference on Artificial Intelligence, New Orleans, LA, USA, February 2018 [1]. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. E. Ricci. *(Corresponding author: Song Wang.)*

Hongkai Yu is with the Department of Electrical Engineering and Computer Science, Cleveland State University, Cleveland, OH 44115 USA (e-mail: h.yu19@csuohio.edu).

Kang Zheng and Hao Guo are with the Department of Computer Science & Engineering, University of South Carolina, Columbia, SC 29208 USA (e-mail: zheng37@email.sc.edu; hguo@email.sc.edu).

Jianwu Fang is with the School of Electronic and Control Engineering, Chang'an University, Xi'an 710064, China (e-mail: j.w.fangit@gmail.com).

Song Wang is with the Department of Computer Science & Engineering, University of South Carolina, Columbia, SC 29208 USA, and also with the School of Computer Science and Technology, Tianjin University, Tianjin 300072, China (e-mail: songwang@cec.sc.edu).

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

Digital Object Identifier 10.1109/TMM.2020.2972165

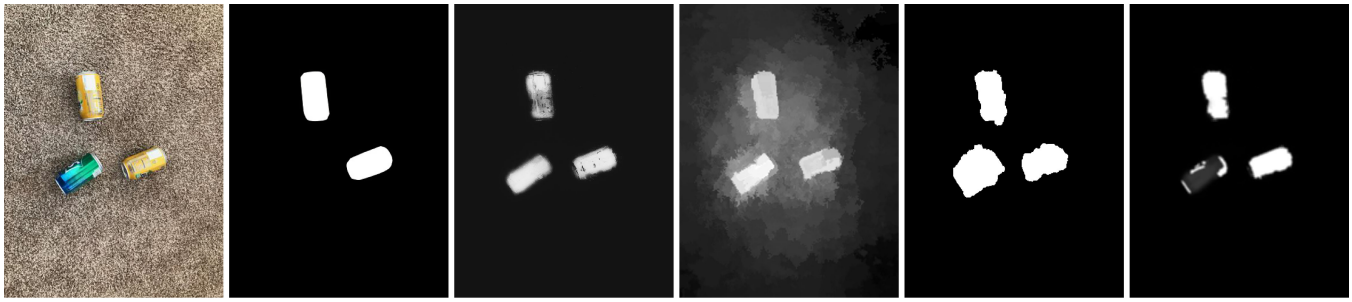


Fig. 2. Illustrations for the difficulties of applying existing methods to solve the proposed within-image co-saliency problem. (a) Original image. (b) Ground truth. (c) Result by the within-image saliency method [18]. (d) Result by the cross-image co-saliency method [19] after making two copies of the original image. (e) Result by the repetitive pattern detection method [20]. (f) Result by the proposed method.

an object class in an image and help estimate the number of instances of the same object class [21], [22]. By combining the features of the identified co-salient objects, we may obtain more accurate and more reliable object recognition and detection in the image. Within-image co-saliency detection can also help identify and reduce information redundancy within an image. For example, recent mobile plant-recognition systems [23] usually require the user to take an plant image using his/her smart phone camera and then send the plant image to a remote server for large-scale plant-species classification. The proposed within-image co-saliency detection can identify multiple instances of the same plant part, e.g., leaf, and then crop out only one of them before sending it to the remote server. This may substantially reduce the data size and communication load, which may be greatly helpful for multimedia encoding and transmitting. As in [24], repeated instances of an object class can be used to synthesize realistic animation from a still picture for graphics applications, which could be helped by within-image co-saliency detection.

However, within-image co-saliency detection is a nontrivial problem. As far as we know, there is no existing work that explicitly discusses and tackles this problem. None of existing methods could be directly applied to solve the proposed new within-image co-saliency problem as shown in Fig. 2. On one hand, this problem cannot be well addressed by directly applying the existing methods on saliency detection. By using a within-image saliency detection method, we may also highlight objects that show good saliency but not co-saliency, e.g., the green soda can in Fig. 2(c). On the other hand, cross-image co-saliency methods are also not applicable here because we only have one input image. One naive solution might be making multiple copies of the input image and then applying a cross-image co-saliency detection method. However, this solution still does not work, because it will highlight all the salient objects in the image as shown in Fig. 2(d). For example, if we make two copies of the original image and then apply a cross-image co-saliency detection algorithm, all three soda cans including the green soda can will be highlighted. Furthermore, the methods of detecting the repetitive patterns in a single image will emphasize all the similar-pattern objects in the image, including the green soda can as shown in Fig. 2(e). For the proposed within-image co-saliency detection problem, it wants to highlight the salient objects with common saliency and also de-emphasize the objects without common saliency as shown in Fig. 2(f).

In this paper, we propose a new bottom-up method for detecting within-image co-saliency. Given an image, we first detect a large number of object proposals [25]. We then develop an optimization algorithm to derive a set of *proposal groups*, each of which consists of multiple selected proposals showing good common saliency in the original image. Three factors are considered in measuring the common saliency for a group of proposals: 1) the saliency of each proposal in the original image, 2) similar image appearance of the proposals, and 3) low spatial overlap of the proposals. For each derived proposal group, a co-saliency map is computed by a clustering-based algorithm. We then fuse the co-saliency maps computed from different proposal groups into a final co-saliency map using a low-rank based algorithm. Since most existing image datasets used for saliency detection do not consider the within-image co-saliency, we collect a new benchmark dataset of 664 images (two subsets) for performance evaluation. In the experiment, we test the proposed method and other comparison methods on the new dataset and quantitatively evaluate their performance based on the annotated ground truth.

This paper is the first systematical study to the new research problem of within-image co-saliency detection. Our main contributions in this paper can be summarized as follows: 1) We propose and introduce the new research problem of within-image co-saliency to the computer vision and multimedia communities. 2) We propose an effective bottom-up algorithm using convex optimization and low-rank fusion to solve this new problem. 3) We collect a benchmark dataset of 664 color images (two subsets) to evaluate algorithms for this new problem. 4) We show that the proposed method can be also applied to detect the repetitive patterns in a single image or detect the co-saliency in multiple images. A preliminary version of this work has been published in a conference proceeding [1]. Compared to [1], this journal paper enlarges the benchmark dataset (nearly double-sized), displays more properties and constructions of the proposed benchmark dataset, shows more experimental results on the enlarged benchmark, and discusses more details about the algorithm performance, such as performance analysis with different initials, failure case and running time,

and shows the applications of the proposed method in detecting the repetitive patterns in a single image and detecting the co-saliency in multiple images.

The remainder of the paper is organized as follows. Section II overviews the related work. Section III introduces the proposed method on within-image co-saliency detection. Section IV reports the benchmark dataset and experimental results, followed by a brief conclusion in Section V.

II. Related Work

As mentioned above, most previous work on image-based saliency detection is focused on two problems: saliency detection in a single image, i.e., within-image saliency detection, and co-saliency detection in multiple images, i.e., cross-image co-saliency detection.

Many within-image saliency detection models and methods have been developed in the past decades. Most traditional methods identify salient regions in an image based on visual contrasts [6]. Many hand-crafted rules, such as center bias [12], frequency [26], and spectral residuals [27] have been incorporated to improve the saliency detection performance. Graph-based segmentation algorithms [28], [29] could be applied to refine the resulting saliency maps [6]. In [30]–[33], high-level knowledges such as objectness, fixation predictions, object boundary, and low rank consistency are integrated to achieve within-image saliency detection, besides the use of low-level features like color, texture and SIFT features. Recently, deep learning techniques have also been used for detecting saliency in an image by automatically learning the features. In particular, it has been shown that the multi-scale deep learning [34] using patch-level convolutional neural networks (CNN) [35], the deep contrast learning [18] with pixel-level fully convolutional networks (FCN) [36] and the recurrent fully convolutional networks (RFCN) [37] can detect the within-image saliency more accurately than many of the above-listed traditional methods.

Cross-image co-saliency detection has also been studied by many researchers recently. In [12], [38], each pixel is ranked by using manually designed co-saliency cues such as inter-image saliency cue, intra-image saliency cue, and repeatedness cue. In [39]–[41], co-saliency maps produced by different methods are fused by further exploring the inter-image correspondence. Recently, machine learning based methods like weakly supervised learning [42], multiple instance learning [43], and deep learning [13], [44] are also used for cross-image co-saliency detection. Depth cues together with the RGB information (RGBD images) can be also used to help the cross-image co-saliency detection [45]–[47]. Other problems related to cross-image co-saliency detection are co-localization [16], [17] and co-segmentation [5], [15], [48], which aim to localize or segment common objects that are present in multiple input images. However, all these within-image saliency detection and cross-image co-saliency detection methods cannot address the problem of within-image co-saliency detection, on which this paper is focused, because they could not de-emphasize salient objects that do not show within-image co-saliency, e.g., the white-clothed player in Fig. 1(c).

Another work related to our problem is the supervised object detection, a fundamental problem in computer vision. In [49], top-down approaches considering object detection are developed for detecting within-image saliency – objects detected in the image are emphasized in the saliency map. Ideally, we may extend it to within-image co-saliency detection: run an object detector [50], [51] on the given image and then match the detected objects. If two or more detected objects show high-level of similarity and belong to the same object class, we highlight them in the resulting co-saliency map. If a detected object does not match to any other detected object in the image, we de-emphasize it in the resulting co-saliency map. However, object detector can only detect known object classes [52] that are pre-trained using supervised learning, not to mention that highly-accurate large-scale object detection itself is still a challenging research problem. Just like most previous work on saliency detection, in this paper we detect within-image co-saliency without assuming any specific object class and recognizing any objects in the image.

Another work related to our problem is the repetitive pattern detection in a single image [20], [53], [54], which extracts or segments all the objects with similar structures in a single image. It is also different from the proposed research of within-image co-saliency detection in the following two perspectives: 1) The proposed research is based on saliency detection that ignores the repeated patterns in the non-salient regions, i.e., backgrounds; 2) The proposed research also suppresses the salient objects without showing co-saliency in a single image.

III. Proposed Method

The basic idea of the proposed method is to first generate many object proposals (in the form of rectangular bounding boxes) in the image, and then compute co-saliency by deriving proposal groups with good common saliency in the image. The diagram of the proposed method is illustrated in Fig. 3. For object proposals, as mentioned above, we do not consider any prior information on the object classes and they are detected only based on general objectness. Considering the possibility that many detected object proposals do not well cover a real object, as shown in the second column of Fig. 3, we identify different proposal groups where each group of proposals show good common saliency by a convex optimization model. With these found proposals showing good common saliency, we apply the clustering algorithm to find the dominant clusters in the found proposals, and then the dominant clusters are used to define the co-saliency map. The clustering is used to transfer the detected proposal groups by the convex optimization into co-saliency maps. We compute such common saliency for each proposal group in the form of a co-saliency map in the original image and finally fuse the co-saliency maps computed from different proposal groups for the desired within-image co-saliency. Because the object proposals and clustering results are both not perfect, we generate a co-saliency map for each found proposal group. Then, we rely on the low-rank method to fuse the final co-saliency map so as to improve the performance.

In this paper, we use the classical bottom-up EdgeBox method [25] to generate object proposals in the image. More

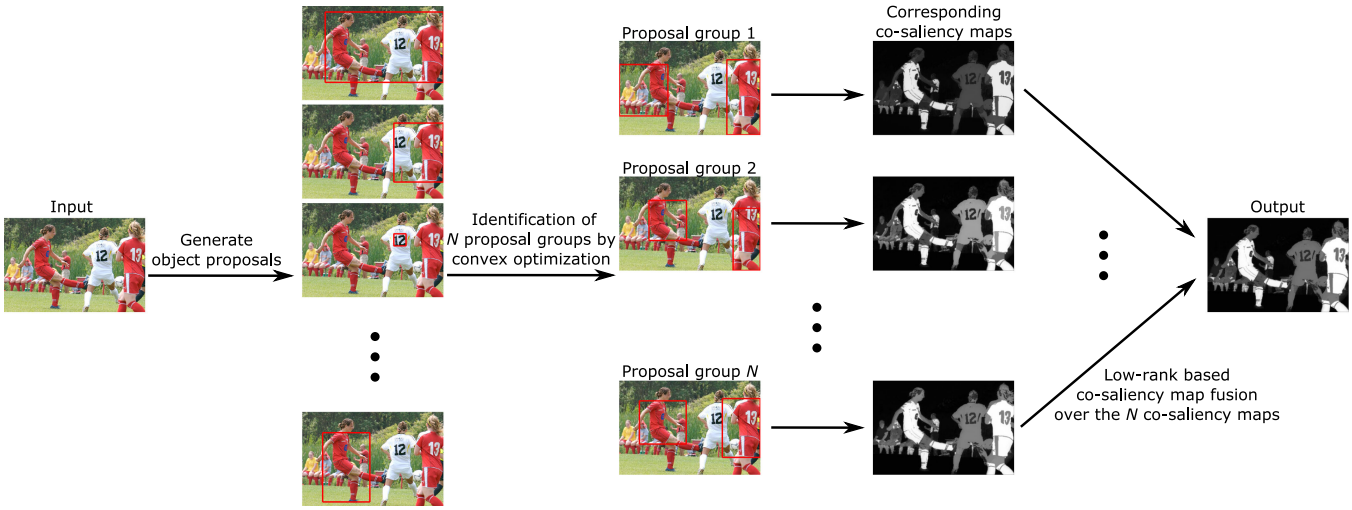


Fig. 3. Diagram of the proposed method for detecting within-image co-saliency.

specifically, we first use EdgeBox to generate a large pool of proposals with different objectness scores. After pruning overly small proposals (with size <1% of the image size), we select M object proposals with the highest objectness scores from the pool and denote them as $P_i, i = 1, 2, \ldots, M$. Based on these M detected proposals, we elaborate on the other three main components of the proposed methods, i.e., identification of proposal groups, computing co-saliency map for a proposal group, and fusion of multiple co-saliency maps, in this section.

A. Identification of Proposal Groups

Given M object proposals $P_i, i = 1, 2, \ldots, M$ in the image, we identify N different proposal groups, each of which consists of a subset of proposals with good common saliency. In this section, we identify these N proposal groups iteratively: After identifying the first proposal group with highest common saliency, we exclude the identified proposals and apply the same algorithm to identify another proposal group. This process is repeated N times to obtain N proposal groups. For simplicity, we fix the number of proposals in each group to be $K > 1$, which is a pre-set constant. In this paper, we consider three main factors in measuring the common saliency of K proposals in a group: 1) saliency of each of these K proposals, 2) high appearance similarity of these K proposals, and 3) low spatial overlap of these K proposals.

A proposal group can be denoted by a vector $\mathbf{z} = (z_1, z_2, \ldots, z_M)^T$, where $z_i \in \{0, 1\}$, with 1 indicating that proposal i is included in the group and 0 otherwise. First, we can use any within-image saliency detection algorithm [6], [12], [18], [55] to compute an initial saliency map $h(X)$, where X represents all the pixels in the input image and $h(\mathbf{x})$ is the saliency value at pixel $\mathbf{x} \in X$. The saliency of each proposal P_i is the mean saliency value in that proposal region, which can be estimated as $h_i = \frac{1}{|P_i|} \sum_{\mathbf{x} \in P_i} h(\mathbf{x})$. The saliency of all M proposals can be summarized into a column vector $\mathbf{h} = (h_1, h_2, \ldots, h_M)^T$. Following [17], [56], we define a *saliency energy term* to reflect the total saliency of a proposal group $\mathbf{z}$ in the original image by

$$E_1(\mathbf{z}) = -\mathbf{z}^T \log(\mathbf{h}). \tag{1}$$

The smaller this energy term, the larger the saliency of this proposal group in the original image.

To consider the high appearance similarity and low spatial overlap of the proposals in a group $\mathbf{z}$, we first define a pairwise similarity between two proposals, say P_i and P_j, as

$$w_{ij} = \frac{1}{d_{ij}^2 + o_{ij}^2}, \tag{2}$$

where d_{ij} is the L_2 distance between the appearance features of P_i and P_j, and o_{ij} reflects the spatial overlap of P_i and P_j. Specifically, we compute the appearance feature of a proposal by using the normalized RGB color histogram (256×3 bins) of all the pixels in the proposal. We define o_{ij} as $\frac{|P_i \cap P_j|}{\min(|P_i|, |P_j|)}$.

Based on the pairwise similarity w_{ij}, we construct a similarity matrix $\mathbf{W} = (w_{ij})_{M \times M}$. $\mathbf{W}$ is a symmetric matrix and we set all diagonal element w_{ii} to be 0. The normalized Laplacian matrix can then be computed by $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} \mathbf{W} \mathbf{D}^{-\frac{1}{2}}$, where $\mathbf{I}$ is an $M \times M$ identity matrix, $\mathbf{D}$ is the degree matrix, i.e., a diagonal matrix, whose i-th diagonal element takes the value of $\sum_{j=1}^{M} w_{ij}$. Using $\mathbf{L}$, we can define a *similarity energy term* for a proposal group $\mathbf{z}$ that encourages high appearance similarity and low spatial overlap as

$$E_2(\mathbf{z}) = \mathbf{z}^T \mathbf{L} \mathbf{z}. \tag{3}$$

Combining the two energy terms shown in Eqs. (1) and (3), we define the following constrained optimization problem for identifying a proposal group:

$$\begin{aligned} \min_{\mathbf{z}} \quad & \mathbf{z}^T \mathbf{L} \mathbf{z} + \lambda(-\mathbf{z}^T \log(\mathbf{h})) \\ \text{s.t.} \quad & z_i \in \{0, 1\}, i = 1, 2, \ldots, M \\ & \sum_{i=1}^{M} z_i = K, \end{aligned} \tag{4}$$

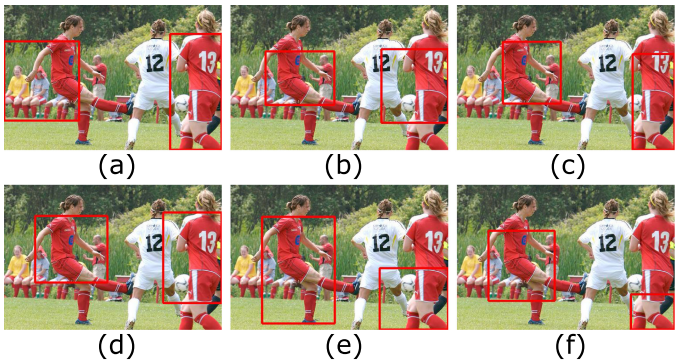


Fig. 4. Six proposal groups identified from a sample image. (a–f) Proposal groups identified from iteration 1 to iteration 6, respectively. Here we set $K = 2$. Note that the object proposals are not perfect to cover each entire object.

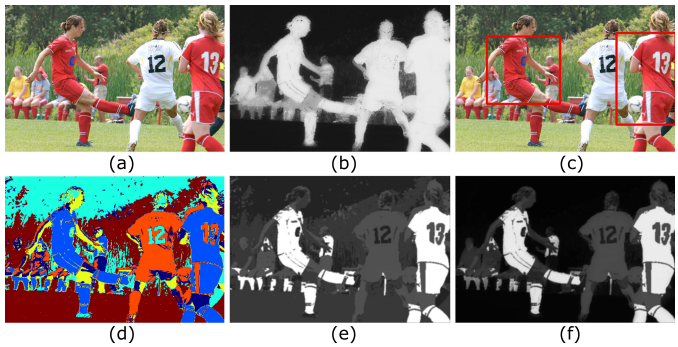


Fig. 5. An example of co-saliency detection triggered by a proposal group. (a) Original image, (b) initial saliency map $h(X)$ [18], (c) a proposal group $\mathcal{P}$ with two proposals, (d) Kmeans clustering results, where each color indicates a cluster, in total six clusters, (e) normalized common-saliency map $\hat{\hbar}_{\mathcal{P}}(X)$, and (f) co-saliency map $\hbar_{\mathcal{P}}(X)$.

where $\lambda > 0$ is a balance factor for the two energy terms, and the last constraint indicates that we seek a group of K proposals with high common saliency. Since the optimization variables in $\mathbf{z}$ are binary, this is not a convex optimization problem. To make it convex, we relax the first constraint in Eq. (4) to

$$0 \leq z_i \leq 1, i = 1, 2, \ldots, M.$$

This way, the optimization problem becomes a standard quadratic programming under linear constraints, since the saliency energy term in Eq. (1) is linear and the similarity energy term in Eq. (3) is quadratic. We can solve this problem efficiently using the primal-dual interior-point method by the CVX convex optimization toolbox [57]. After we get the optimal solution $\mathbf{z}$, we simply select the K proposals with the highest values in $\mathbf{z}$ to form a proposal group. As mentioned above, we iterate this optimization algorithm N times to construct N proposal groups. Fig. 4 shows the proposal groups identified from a sample image.

B. Co-saliency Detection in a Proposal Group

Without loss of generality, let $\mathcal{P} = \{P_1, P_2, \ldots, P_K\}$ be an identified proposal group. In this section, we detect the common saliency in this proposal group and summarize this common saliency into a co-saliency map in the original image. Starting from the initial saliency map $h(X)$, we first threshold this saliency map by a threshold (0.2 in our experiments) to obtain salient region X_T. Inspired by previous work on cross-image co-saliency detection [12], we apply the Kmeans algorithm to cluster all the pixels X in the input image into Z clusters $C_1, C_2, \ldots, C_Z$ based on these pixels' RGB color values. If a cluster shows good spatial overlap with the considered proposal group $\mathcal{P}$, the pixels in this cluster tends to show higher within-image co-saliency in the original image.

More specifically, for each pixel $\mathbf{x} \in C_z$, we define its unnormalized common-saliency map value triggered by proposal group $\mathcal{P}$, which consists of proposals $P_1, P_2, \ldots, P_K$, as

$$\hbar'_{\mathcal{P}}(\mathbf{x}) = \frac{|(\cup_{k=1}^{K} P_k) \cap X_T \cap C_z|}{|(\cup_{k=1}^{K} P_k) \cap X_T|}, \tag{5}$$

where the denominator is the number of salient pixels that are located in the proposal group $\mathcal{P}$ and the numerator is the number of salient pixels in cluster C_z that are located in the proposal group $\mathcal{P}$. We then normalize the map $\hbar'_{\mathcal{P}}(X)$ to a standard Gaussian distribution and denote the normalized common-saliency map triggered by proposal group $\mathcal{P}$ as $\hat{\hbar}_{\mathcal{P}}(X)$. To reduce the effect of clustering errors, we further combine the initial saliency map $h(X)$ and the common-saliency map $\hat{\hbar}_{\mathcal{P}}(X)$ by pixel-wise multiplication to construct a co-saliency map $\hbar_{\mathcal{P}}(X)$ as

$$\hbar_{\mathcal{P}}(\mathbf{x}) = \hat{\hbar}_{\mathcal{P}}(\mathbf{x}) \cdot h(\mathbf{x}), \mathbf{x} \in X,$$

followed by thresholding (0.2 in our experiments), holes filling and average filtering. In Fig. 5, we use a sample image to illustrate the process of this co-saliency detection.

C. Co-Saliency Map Fusion

Based on N identified proposal groups, we can use each of them as the trigger to compute a co-saliency map. In this way, we obtain N co-saliency maps, which we denote as $\{\hbar_1(X), \hbar_2(X), \ldots, \hbar_N(X)\}$. In this section, we study how to fuse these N co-saliency maps into a unified co-saliency map.

After simple thresholding, we find that the co-salient regions in the N co-saliency maps display color-feature consistency when mapped back to the original color image, where the color-feature consistency could be thought as a low rank constraint. Meanwhile other salient objects but not showing within-image co-saliency and the background are treated as sparse noises. In this paper, we adapt the method in [39] for fusing the N co-saliency maps. First, for each co-saliency map, say $\hbar_i(X)$, we apply a simple thresholding as that in [39] to get pixels with high co-saliency. We then compute the RGB color histogram (1,000 bins) of all the identified pixels with high co-saliency by mapping back to the original color image and denote this histogram as a column vector $\mathbf{f}_i$. Combining the N histograms computed from N co-saliency maps, respectively, we obtain a feature matrix $\mathbf{F} = (\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_N)$. We then seek to recover a low-rank matrix $\mathbf{R}$ from $\mathbf{F}$, i.e.,

$$(\mathbf{R}^*, \mathbf{E}^*) = \arg\min_{\mathbf{R},\mathbf{E}} \ (rank(\mathbf{R}) + \beta \|\mathbf{E}\|_0)$$
$$\text{s.t. } \mathbf{F} = \mathbf{R} + \mathbf{E}, \tag{6}$$

Algorithm 1: Co-Saliency Detection Within a Single Image

Input: A color image
1: Use EdgeBox [25] to generate M object proposals.
2: Compute the initial saliency map $h(X)$.
3: Generate Z clusters by Kmeans algorithm.
4: **FOR** $i = 1 : N$
5: Identify i-th proposal group by solving Eq. (4).
6: Compute co-saliency map $\hbar_i(X)$.
7: Exclude proposals in the i-th proposal group.
8: Update the similarity matrix $\mathbf{W}$.
9: **END FOR**
10: Fuse the N co-saliency maps $\hbar_i(X), i = 1, 2, \ldots, N$ for the final co-saliency map $\hbar(X)$.

where $\beta > 0$ is a balance factor between the rank of $\mathbf{R}$ and the L_0 norm of the sparse noise $\mathbf{E}$. By using nuclear norm to approximate $rank(\mathbf{R})$ and L_1 norm to approximate $\|\mathbf{E}\|_0$, this low-rank matrix recovery problem becomes convex and can be solved by robust PCA [58].

Following [39], the final fused co-saliency map can be written as a weighted linear combination of the N co-saliency maps, i.e.,

$$\hbar(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i \cdot \hbar_i(\mathbf{x}), \mathbf{x} \in X, \tag{7}$$

where the weight α_i can be calculated by

$$\alpha_i = \frac{\exp(-\|\mathbf{e}_i^*\|_2)}{\sum_{i=1}^{N} \exp(-\|\mathbf{e}_i^*\|_2)}. \tag{8}$$

In Eq. (8), $\mathbf{e}_i^*$ is the i-th column of $\mathbf{E}^*$ resulting from Eq. (6). Less sparse noise $\mathbf{e}_i^*$ indicates that the i-th co-saliency map $\hbar_i(X)$ is more credible and it should be weighted more in computing the final co-saliency map $\hbar(X)$. The entire proposed method for detecting co-saliency within a single image is summarized in Algorithm 1.

IV. Experiments and Benchmark

Existing publicized image datasets for evaluating saliency detection such as MSRA [59], PASCAL-S [31], HKU-IS [34], iCoseg [60] are mainly collected for testing within-image saliency detection or cross-image co-saliency detection methods. In most cases, each image only contains one salient object, which is annotated as the ground truth. In this paper, we have a different goal of detecting within-image co-saliency, which is not shown in most images in the publicized datasets. Therefore, we collect a new benchmark dataset (named CDS) of 664 color images for Co-saliency Detection within a Single image, consisting of two subsets. The first subset named CDS_1 has 364 color images and the second subset named CDS_2 has 300 color images. Each image in the proposed benchmark shows certain level of within-image co-saliency, e.g., the presence of multiple instances of the same object class with very similar appearance. In this paper, we **define** that an image showing within-image co-saliency also containing salient object(s) without showing any within-image co-saliency is a *challenging image*, and an image only showing within-image co-saliency is an *easy image*. In the 364 images of CDS_1, about 18% images are *challenging* while other 82% images are *easy*. In the 300 images of CDS_2, most images (99%) are *challenging* images. The percentage of challenging images in two subsets and example *easy* and *challenging* images are shown in Fig. 6, which also shows the histogram of the image number of each object class in the whole benchmark dataset. The images of the proposed benchmark dataset are from the object classes of Animal, Human, Fruit, Food, Airplane, Vehicle, Flower and Other, where 'Other' mainly means the object classes of different indoor items like shoes, sodas, napkins, balls, cans, etc. More example images and their corresponding manually labeled ground truth can be found in Fig. 7.

In CDS_1, 100 images are selected from the iCoseg [60], MSRA [59], HKU-IS [34] datasets and the remaining images are collected from the Internet. For CDS_1, the image size ranges from 150×150 to 808×1078 pixels. The size of images in CDS_2 ranges from 169×298 to 504×378 pixels. In CDS_2, 35 images are selected from the Internet and the remaining images are taken by ourselves in the indoor environment. Co-salient objects within each image of the two subsets are manually labeled as the ground truth (a binary mask) for performance evaluation. To avoid unreasonable labeling, the ground truths are double checked by five different researchers in computer vision area. Because the two subsets CDS_1 and CDS_2 have different percentages of *challenging* images, we conduct experiments on the two subsets independently. Compared to [1], the preliminary version [1] only used CDS_1 for evaluation and CDS_2 is an added new set for evaluating within-image co-saliency detection.

In our experiment, we generate $M = 100$ object proposals. The number of proposal groups is set to $N = 10$. The number of proposals in each group is set to $K = 2$. We set the balance factors $\lambda = 0.01$ in Eq. (4) and $\beta = 0.05$ in Eq. (6). The number of clusters is set to $Z = 6$ in the Kmeans algorithm. The initial within-image saliency map $h(X)$ is computed using the algorithm developed in DCL [18]. Seven state-of-the-art within-image saliency detection methods are chosen as the comparison methods: CWS [12], LRK [33], SR [27], FT [26], RC [6], DCL [18], and RFCN [37]. The first five are traditional feature-based methods and the last two are based on deep learning.

As in many previous works [6], [12], [18], [26], we evaluate the performance using precision-recall (PR) curve, maximum F-measure (maxF), MAE error and also report the average precision, recall and F-measure using an adaptive threshold. The resulting saliency map can be converted to a binary mask with a threshold, and the precision and recall are computed by comparing the binary mask and the binary ground truth. Varying the threshold continuously in the range of [0, 1] leads to a PR curve, which is averaged over all the images in the dataset in this paper. As in [18], we can calculate the maximum F-measure (maxF) from the PR curve and the MAE error as the average absolute per-pixel difference between the resulting saliency map and the ground truth. As in [18], [26], we also use an adaptive threshold, i.e., twice the mean value of the saliency map, to convert the saliency map into a binary mask. Comparing the binary mask with the binary ground truth, we can compute the precision and recall, based on which we can compute F-measure

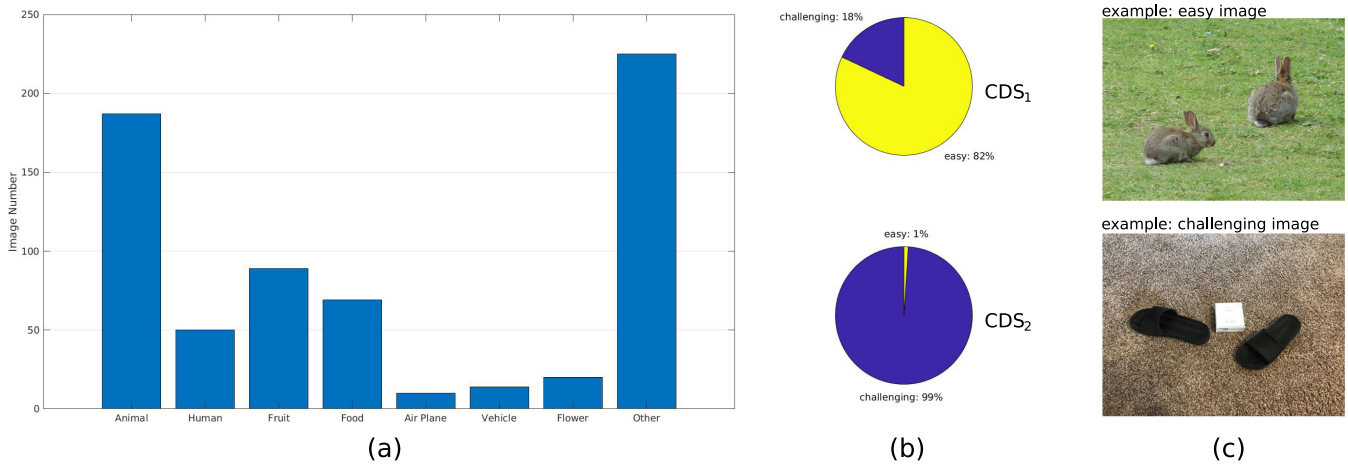


Fig. 6. The proposed benchmark dataset (664 images): (a) histogram of the image number of each object class in the whole benchmark dataset, where 'Other' mainly means the object classes of different indoor items like shoes, sodas, napkins, balls, cans, etc; (b) percentage of *challenging* images in the subset CDS_1 (364 images) and in the subset CDS_2 (300 images); (c) example of an *easy* image and a *challenging* image based on our definition.

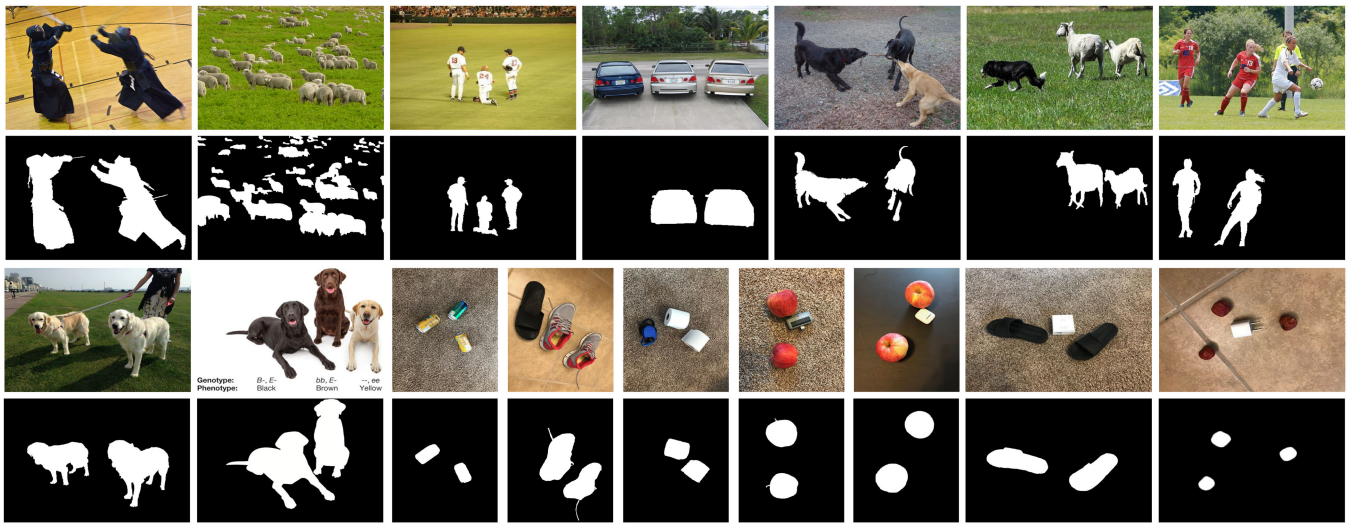


Fig. 7. Sample images and their corresponding within-image co-saliency ground truth in the proposed benchmark dataset. The top two rows are from the subset CDS_1 (first three columns: *easy*, last four columns: *challenging*). The bottom two rows are from the subset CDS_2 (all columns: *challenging*).

as $F_\gamma = \frac{(1+\gamma^2)\times Precision\times Recall}{\gamma^2\times Pecision+Recall}$, where γ^2 is set to 0.3 as defined in [12], [18], [26].

A. *Experimental Result on the Subset* CDS_1

In this section, we will introduce the experimental results on the subset CDS_1. Fig. 8 shows the PR curves of the proposed method and seven comparison methods that were developed for within-image saliency detection. Table I compares the maxF and MAE error of the proposed method against these seven comparison methods. We can see that the proposed method achieves the best performance in detecting within-image co-saliency in terms of these evaluation metrics on the subset CDS_1.

The average precision, recall and F-measure using adaptive thresholds [18], [26] are shown as a bar chart in Fig. 9. We

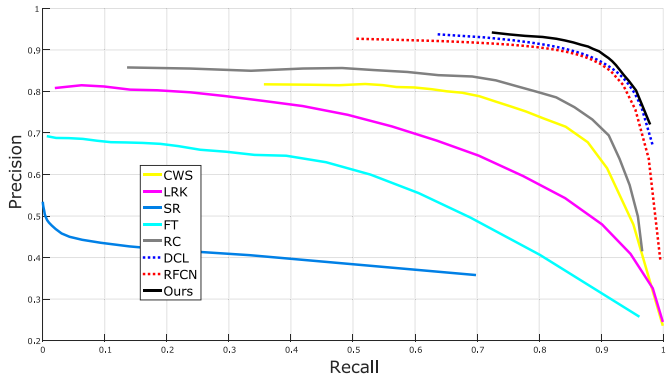


Fig. 8. PR curves of the proposed method ('Ours') and the seven saliency detection methods, averaged over all the 364 images on CDS_1.

TABLE I
THE MAXIMUM F-MEASURE (MAXF) AND MAE ERROR OF THE PROPOSED METHOD ('OURS') AND THE SEVEN WITHIN-IMAGE SALIENCY DETECTION METHODS ON CDS_1. LARGER MAXF AND SMALLER MAE ERROR INDICATE BETTER PERFORMANCE

Metric	CWS	LRK	SR	FT	RC	DCL	RFCN	Ours
maxF (%)	76.7	67.4	40.3	58.2	80.2	88.8	88.3	**90.3**
MAE error	0.165	0.241	0.246	0.244	0.142	0.059	0.083	**0.050**

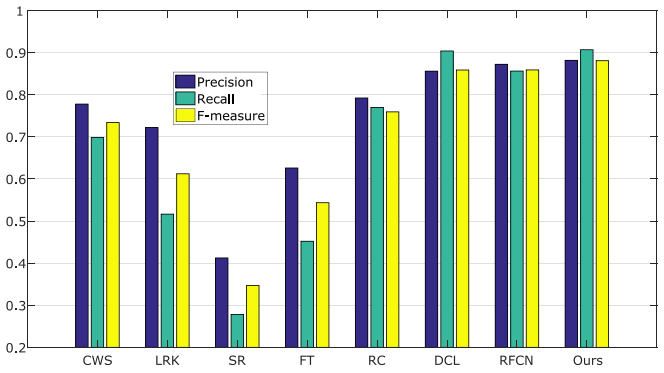


Fig. 9. Average precision, recall and F-measure using adaptive thresholds on CDS_1. Our F-measure: 88.1%; second best F-measure: 85.9%.

can see that, using adaptive thresholds, the proposed method achieves the best average precision, recall and F-measure against the seven comparison methods in detecting within-image co-saliency. Specifically, the average precision, recall and F-measure using adaptive thresholds are [0.882, 0.907, 0.881] when using the proposed method, while the second best is achieved by DCL [18] ([0.856, 0.904, 0.859]) and RFCN [37] ([0.872, 0.856, 0.859]).

Fig. 10 shows sample results of within-image co-saliency detection from the proposed method and the comparison methods including seven within-image saliency detection methods on the subset CDS_1.

B. Experimental Result on the Subset CDS_2

In this section, we will introduce the experimental results on the subset CDS_2. Fig. 11 shows the PR curves of the proposed method and seven comparison methods that were developed for within-image saliency detection. Table II compares the maxF and MAE error of the proposed method against these seven comparison methods. We can see that the proposed method achieves the best performance in detecting within-image co-saliency in terms of these evaluation metrics on the subset CDS_2.

The average precision, recall and F-measure using adaptive thresholds [18], [26] are shown as a bar chart in Fig. 12. Using adaptive thresholds, the proposed method achieves the best average precision, recall and F-measure against the seven comparison methods in detecting within-image co-saliency. Specifically, the average precision, recall and F-measure using adaptive thresholds are [0.737, 0.954, 0.773] when using the proposed method, while the second and third bests are achieved by DCL [18] ([0.716, 0.929, 0.752]) and RFCN [37] ([0.637, 0.942, 0.682]). Fig. 13 shows sample results of within-image co-saliency detection from the proposed method and the comparison methods including seven within-image saliency detection methods on the subset CDS_2.

C. Experimental Result Summary on the Benchmark

Based on the experimental results on CDS_1 and CDS_2, in general, the proposed method performs better than all these seven comparison methods in detecting the within-image co-saliency. We can also see that the two deep learning based methods (DCL [18], RFCN [37]) can detect better within-image co-saliency than the five traditional saliency detection methods. Among the five traditional methods, CWS [12] and RC [6] show relatively better performance in detecting within-image co-saliency. We see that the proposed method is capable of highlighting the regions that show within-image co-saliency and de-emphasizing the salient regions that do not show within-image co-saliency. However, the comparison methods might highlight all the salient regions or ignore to emphasize the regions that show within-image co-saliency.

Because CDS_2 contains more percentage of *challenging* images than CDS_1, the evaluation performance on CDS_2 is lower than that of CDS_1. Taking the PR curve as an example, we see that the advantages of the proposed method over other comparison methods are more significant on the subset CDS_2 that contains more *challenging* images. Specifically, the maxF of 'Ours' is 90.3% and the second best is 88.8% by DCL on CDS_1, while the maxF of 'Ours' is 86.6% and the second best is only 79.1% by DCL on CDS_2.

D. Experimental Results With Different Initials

The proposed method starts with an initial within-image saliency map $h(X)$, as described in Section III-A. In the above experiments, we use DCL for computing $h(X)$. Many other methods can also be used to compute $h(X)$. We conduct an experiment on the subset CDS_2 by using different initial saliency maps and examine its influence to the performance of the proposed method. Using the PR curve, maxF from different $h(X)$, computed by DCL, RFCN, RC, and CWS, respectively, are reported in Table III. The second row of this table is the maxF by directly comparing the map $h(X)$ and the ground truth. The third row is the maxF of the proposed method initialized by the corresponding method. We can see that the co-saliency map resulting from the proposed method is always better than the used initial saliency map $h(X)$, with significant improvements. We can also see that, using a better initial saliency map $h(X)$, e.g., higher maxF in the second row of Table III, can improve the performance of the proposed method. Fig. 14 shows the results of the proposed method on an example image using different initial saliency maps.

E. Parameter Effects and Low-Rank Fusion Effects

Parameter effects: We use the subset CDS_1 to discuss the effects of different parameter setting for the proposed method. We run experiments for different M (100, 200, 300) in the proposed method, and the standard deviation of final maxF for the proposed method with different proposal numbers M is only 0.3%. We also test the proposed method using different cluster

Fig. 10. Within-image co-saliency detection results on sample images of CDS_1.

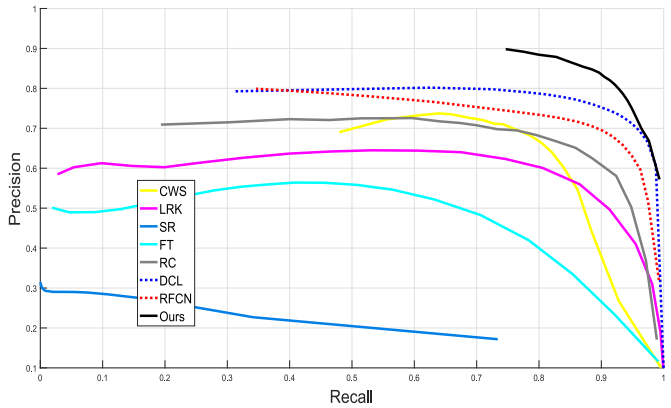


Fig. 11. PR curves of the proposed method ('Ours') and the seven saliency detection methods, averaged over all the 300 images on CDS_2.

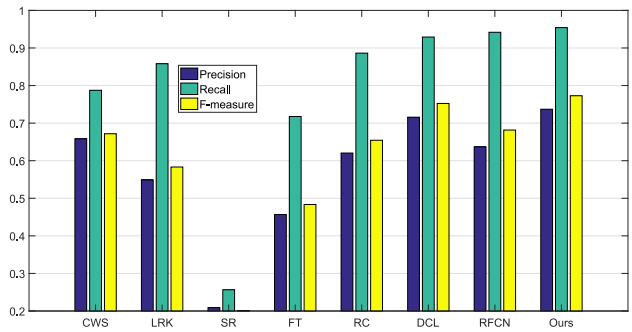


Fig. 12. Average precision, recall and F-measure using adaptive thresholds on CDS_2. Our F-measure: 77.3%; second best F-measure: 75.2%.

TABLE II
THE MAXIMUM F-MEASURE (MAXF) AND MAE ERROR OF THE PROPOSED METHOD ('OURS') AND THE SEVEN WITHIN-IMAGE SALIENCY DETECTION METHODS ON CDS_2. LARGER MAXF AND SMALLER MAE ERROR INDICATE BETTER PERFORMANCE

Metric	CWS	LRK	SR	FT	RC	DCL	RFCN	Ours
maxF (%)	71.8	64.8	25.0	55.1	71.0	79.1	74.8	**86.6**
MAE error	0.121	0.132	0.238	0.123	0.086	0.112	0.060	**0.031**

numbers Z (4, 6, 8) in the proposed method, the standard deviation of final maxF for the proposed method with different Z is only 0.5%. These experiments show that the proposed method is robust with different M and Z. We set $K = 2$ in all the experiments because $K = 2$ is the minimum number to define the co-saliency in a single image. For example, an image shows within-image co-saliency only when it has at least two similar salient objects within the image.

Low-rank fusion effects: For the low-rank based co-saliency map fusion, we follow the same parameter setting in [39]. We also conduct an experiment to show the proposed method without the low-rank fusion on the subset CDS_1, it drops by 0.6% for final maxF, which shows the effectiveness of the low-rank based co-saliency map fusion.

F. Failure Case and Running Time

Failure case: A failure case of the proposed method is shown in Fig. 15. The bridge in the middle of the image contains many white components. These components share very similar appearance and their appearance is also very similar to the two co-salient boats on the left and right sides of the image. The proposed method fails to de-emphasize these components in the final co-saliency map. In our experiments, we also found that, if the initial saliency map $h(X)$ is poor, it will negatively affect the performance of the proposed method on within-image co-saliency detection.

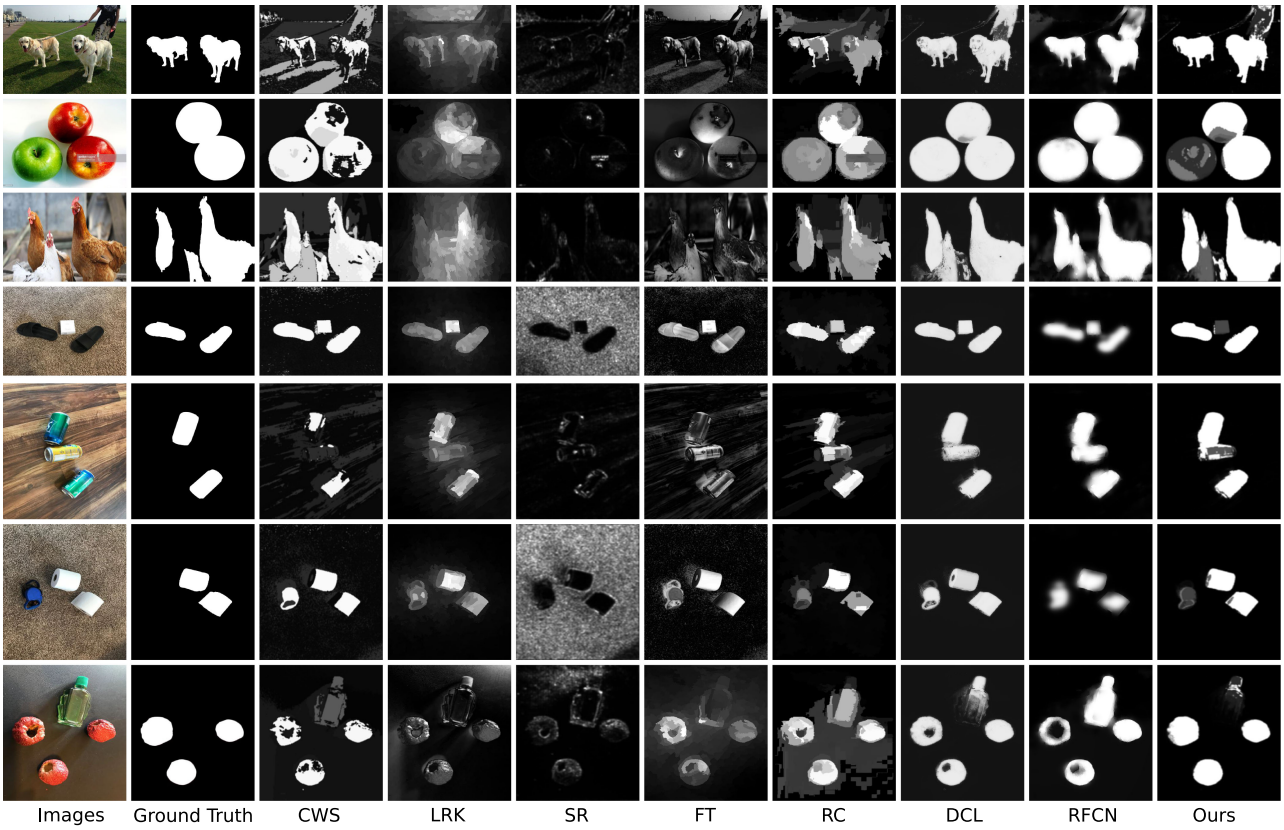


Fig. 13. Within-image co-saliency detection results on sample images of CDS_2.

TABLE III
PERFORMANCE OF THE PROPOSED METHOD WHEN USING DIFFERENT METHODS TO COMPUTE THE INITIAL SALIENCY MAP $h(X)$ (USING CDS_2 AS AN EXAMPLE). WE CAN SEE SIGNIFICANT IMPROVEMENTS BY THE PROPOSED METHOD USING DIFFERENT INITIALS

maxF (%)	DCL	RFCN	RC	CWS
Initial saliency map $h(X)$	79.1	74.8	71.0	71.8
Ours	**86.6**	**82.6**	**78.3**	**76.1**

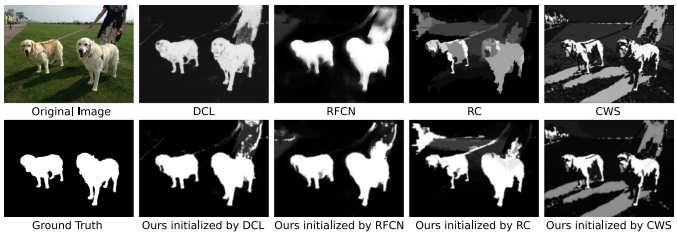


Fig. 14. Results of the proposed method on an example image using different initial saliency maps.

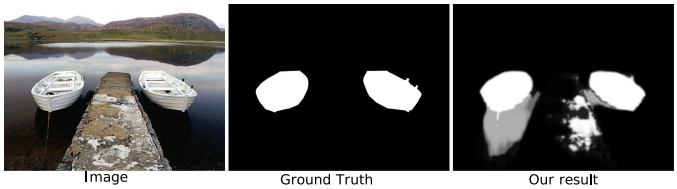


Fig. 15. A failure case of the proposed method.

Fig. 16. Sample results of detecting the repetitive patterns in a single image using two example images in the "repetitive" category of the public Coseg-Rep [53] dataset: (a) an example image, (b) result by the proposed method, (c) another example image, and (d) result by the proposed method.

Running time: For the running time, we implemented the proposed method in Matlab without deliberately optimizing the efficiency. Taking a 360×480 image as an example, on a PC with 3.2 GHz CPU, it takes 11.0 seconds to run the proposed method on this image. Specifically, 2% of time is spent to generating object proposals, 74% of time is spent on identifying the 10 proposal groups, 13% of time is spent on generating the 10 co-saliency maps, clusters and the initial saliency map $h(X)$ (using DCL as example), and the remaining 11% of time is spent on co-saliency map fusion. We expect that this running time can be substantially reduced by optimizing the code and using C++ instead of Matlab.

G. Application of Detecting Repetitive Patterns in a Single Image

In this section, we show that the proposed method could be applied to detect the repetitive patterns in a single image

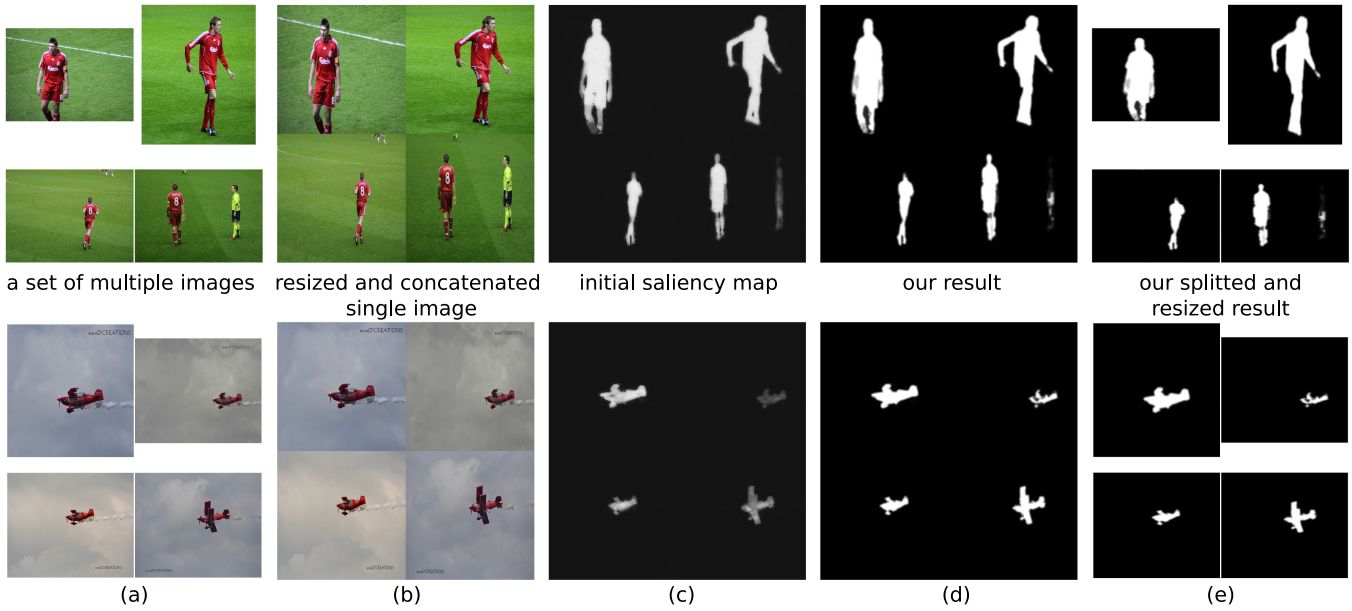


Fig. 17. An illustration of extending the proposed method to detect the cross-image co-saliency using two example classes (Player, Airshow) in the public iCoseg dataset (4 images for each class): (a) a set of multiple images, (b) each image is resized to 300 × 300 and then concatenated into a single image (600 × 600) because the proposed method requires a single image as the input, (c) initial saliency map for (b) by DCL, (d) our result by the proposed method, and (e) our result after splitting and resizing. Note that the yellow-clothed person should be suppressed in the Player class and the airplanes should be emphasized in the Airshow class based on the definition of the cross-image co-saliency.

TABLE IV
PERFORMANCE OF DETECTING THE REPETITIVE PATTERNS IN A SINGLE IMAGE USING THE "REPETITIVE" CATEGORY (116 IMAGES) OF THE PUBLIC COSEGREP DATASET IN TERMS OF ACC (CORRECTLY LABELED PIXEL RATIOS) AND JACCARD (INTERSECTION-OVER-UNION SCORES). OURS$_R$ AND OURS$_D$ INDICATE THE PROPOSED METHOD INITIALIZED BY RFCN AND DCL RESPECTIVELY

Metrics	[52]	[53]	RFCN	DCL	Ours$_{RFCN}$	Ours$_{DCL}$
Acc (%)	86.2	87.9	90.4	91.4	90.7	**91.6**
Jaccard (%)	75.4	**78.2**	75.3	77.3	75.6	77.7

using the "repetitive" category (116 images) of the public CosegRep [53] dataset. In this publicized "repetitive" dataset, each image contains similar patterns repeating themselves within the same image. From the Fig. 16 and Table. IV, we can see that the proposed method obtains reasonable results in detecting the repetitive patterns in a single image, which is comparable with [53] and [54] in terms of Acc (correctly labeled pixel ratios) and Jaccard (Intersection-over-union scores). For a fair evaluation with the comparison methods, we apply the GrabCut segmentation method [28] using each image's saliency map to obtain each image's final binary segmentation mask for the evaluation.

We also conducted another experiment to run [20] on our proposed CDS_2 benchmark dataset (300 images). [20] is a co-segmentation method that can be applied to a single image to extract/segment the repetitive structures in the single image. After applying [20] to each image independently on CDS_2, [20] obtained maxF 0.647, MAE error 0.076, while our proposed method achieved much better maxF 0.866, MAE error 0.031.

This section shows that the proposed method is able to detect repetitive structures in the single image with comparable performances. However, because the within-image co-saliency detection also suppresses the objects without showing common saliency in a single image and ignores the repeated patterns in the non-salient image regions (backgrounds), these two research problems are different.

H. Application of Detecting Co-Saliency in Multiple Images

In this section, we show that the proposed method could be applied to detect the co-salient objects in multiple images. Two kinds of previous research problems are related: cross-image co-saliency detection [12] and cross-image co-segmentation [19], [20], both of which highlight (detect or segment) the co-salient objects in multiple images. We conducted an experiment to compare the proposed method with other cross-image co-saliency or co-segmentation methods. We use an adaptive threshold (two times the mean saliency value as that in [18], [26]) to get the segmentation results from the saliency map. We choose the public iCoseg [60] dataset for this experiment for its popularity in the area of cross-image co-saliency and co-segmentation. The proposed method's input is a single image, while the cross-image co-saliency or co-segmentation method' input is a set of multiple images. With a simple image concatenation, the proposed method can be extended to solve the problem of cross-image co-saliency or co-segmentation. Because the minimum image number of one class in the iCoseg dataset is four, we consistently choose four images in each class of the iCoseg dataset for the experiment. Given multiple images for one class in the iCoseg dataset, we resize each image to a uniform size and concatenate them into a single image as shown in the Fig. 17. For other methods, we apply their publicized codes to the same images of each class as ours in the iCoseg dataset for fair comparisons. The results are summarized in the Table V. It demonstrates that the proposed method can be extended to solve the problems of

TABLE V
PERFORMANCE OF THE CROSS-IMAGE CO-SALIENCY ON THE iCoseg DATASET IN TERMS OF ACC (CORRECTLY LABELED PIXEL RATIOS), JACCARD (INTERSECTION-OVER-UNION SCORES) AND MAE ERROR. OURS$_R$ AND OURS$_D$ INDICATE THE PROPOSED METHOD INITIALIZED BY RFCN AND DCL RESPECTIVELY

Metrics	[20]	[12]	[19]	RFCN	DCL	Ours$_{RFCN}$	Ours$_{DCL}$
Acc (%)	94.2	88.7	93.4	93.0	95.8	93.5	**95.9**
Jaccard (%)	78.4	59.1	74.9	68.4	**80.1**	71.0	79.7
MAE error	0.058	0.152	0.195	0.095	0.126	0.075	**0.052**

cross-image co-saliency and co-segmentation, leading to comparable performance to other cross-image co-saliency [12] and co-segmentation methods [19], [20]. So far as we know, this is the first time to see that the cross-image co-saliency problem could be also solved by the proposed within-image co-saliency method.

V. CONCLUSION

In this paper, we raised a new problem of detecting co-saliency in a single image, i.e., detecting within-image co-saliency. We developed a new bottom-up method to solve this problem. This method starts with detecting a large number of object proposals in the image, without using any prior information on the object classes. We then developed an optimization model to identify a set of proposal groups, each of which consists of multiple proposals with good common saliency in the original image. Co-saliency is then detected in each proposal group and fused for the final within-image co-saliency map. We collected a new benchmark dataset of 664 images including two subsets with good within-image co-saliency, and then used them to test the proposed method. Experimental results showed that the proposed method outperforms the recent state-of-the-art saliency detection methods in detecting within-image co-saliency. The experimental results also displayed that the proposed method can be applied to detect the repetitive patterns in a single image or detect the co-saliency in multiple images.

REFERENCES

[1] H. Yu *et al.*, "Co-saliency detection within a single image," in *Proc. AAAI Conf. Artificial Intell.*, 2018, pp. 7509–7516.

[2] Y. Gao, M. Shi, D. Tao, and C. Xu, "Database saliency for fast image retrieval," *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 359–369, Mar. 2015.

[3] M. Xu, Y. Ren, Z. Wang, J. Liu, and X. Tao, "Saliency detection in face videos: A data-driven approach," *IEEE Trans. Multimedia*, vol. 20, no. 6, pp. 1335–1349, Jun. 2018.

[4] R. Cong *et al.*, "HSCS: Hierarchical sparsity based co-saliency detection for rgbd images," *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1660–1671, Jul. 2019.

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

[6] M.-M. Cheng, N. Mitra, X. Huang, P. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 37, no. 3, pp. 569–582, Mar. 2015.

[7] V. Mahadevan and N. Vasconcelos, "Saliency-based discriminant tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2009, pp. 1007–1013.

[8] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2013, pp. 3586–3593.

[9] R. Huang, W. Feng, and J. Sun, "Saliency and co-saliency detection by low-rank multiscale fusion," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2015, pp. 1–6.

[10] X. Lin, Z.-J. Wang, L. Ma, and X. Wu, "Saliency detection via multi-scale global cues," *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1646–1659, Jul. 2019.

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

[12] 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.

[13] D. Zhang, J. Han, C. Li, and J. Wang, "Co-saliency detection via looking deep and wide," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 2994–3002.

[14] R. Huang, W. Feng, and J. Sun, "Color feature reinforcement for cosaliency detection without single saliency residuals," *IEEE Signal Process. Lett.*, vol. 24, no. 5, pp. 569–573, May 2017.

[15] H. Yu, M. Xian, and X. Qi, "Unsupervised co-segmentation based on a new global GMM constraint in MRF," in *Proc. IEEE Int. Conf. Image Process.*, 2014, pp. 4412–4416.

[16] A. Joulin, K. Tang, and L. Fei-Fei, "Efficient image and video co-localization with frank-wolfe algorithm," in *Proc. Eur. Conf. Comput. Vision*, 2014, pp. 253–268.

[17] K. Tang, A. Joulin, L.-J. Li, and L. Fei-Fei, "Co-localization in real-world images," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2014, pp. 1464–1471.

[18] G. Li and Y. Yu, "Deep contrast learning for salient object detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2016.

[19] 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.

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

[21] X. He and S. Gould, "An exemplar-based CRF for multi-instance object segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2014, pp. 296–303.

[22] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vision*, 2014, pp. 740–755.

[23] N. Kumar *et al.*, "Leafsnap: A computer vision system for automatic plant species identification," in *Proc. Eur. Conf. Comput. Vision*, 2012, pp. 502–516.

[24] X. Xu *et al.*, "Animating animal motion from still," *ACM Trans. Graph.*, vol. 27, no. 5, pp. 1–8, 2008.

[25] C. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Proc. Eur. Conf. Comput. Vision*, 2014, pp. 391–405.

[26] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2009, pp. 1597–1604.

[27] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2007, pp. 1–8.

[28] 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, 2004.

[29] H. Yu, Y. Zhou, H. Qian, M. Xian, and S. Wang, "Loosecut: Interactive image segmentation with loosely bounded boxes," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2017, pp. 3335–3339.

[30] K.-Y. Chang, T.-L. Liu, H.-T. Chen, and S.-H. Lai, "Fusing generic objectness and visual saliency for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vision*, 2011, pp. 914–921.

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

[32] Q. Wang, Y. Yuan, P. Yan, and X. Li, "Saliency detection by multiple-instance learning," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 660–672, Apr. 2013.

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

[34] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 5455–5463.

[35] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[36] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 3431–3440.

[37] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Saliency detection with recurrent fully convolutional networks," in *Proc. Eur. Conf. Comput. Vision*, 2016, pp. 825–841.

[38] C. Ge, K. Fu, F. Liu, L. Bai, and J. Yang, "Co-saliency detection via inter and intra saliency propagation," *Signal Process.: Image Commun.*, vol. 44, pp. 69–83, 2016.

[39] 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.

[40] H. Li, F. Meng, B. Luo, and S. Zhu, "Repairing bad co-segmentation using its quality evaluation and segment propagation," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3545–3559, Aug. 2014.

[41] Z. Tan, L. Wan, W. Feng, and C.-M. Pun, "Image co-saliency detection by propagating superpixel affinities," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2013, pp. 2114–2118.

[42] M.-M. Cheng, N. J. Mitra, X. Huang, and S.-M. Hu, "SalientShape: Group saliency in image collections," *Visual Comput.*, vol. 30, no. 4, pp. 443–453, 2014.

[43] D. Zhang *et al.*, "A self-paced multiple-instance learning framework for co-saliency detection," in *Proc. IEEE Int. Conf. Comput. Vision*, 2015, pp. 594–602.

[44] D. Zhang, J. Han, J. Han, and L. Shao, "Cosaliency detection based on intrasaliency prior transfer and deep intersaliency mining," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 6, pp. 1163–1176, Jun. 2016.

[45] R. Cong *et al.*, "Co-saliency detection for RGBD images based on multi-constraint feature matching and cross label propagation," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 568–579, Feb. 2018.

[46] R. Cong *et al.*, "An iterative co-saliency framework for RGBD images," *IEEE Trans. Cybern.*, vol. 49, no. 1, pp. 233–246, Jan. 2019.

[47] R. Cong *et al.*, "Review of visual saliency detection with comprehensive information," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 10, pp. 2941–2959, Oct. 2019.

[48] K. R. Jerripothula, J. Cai, and J. Yuan, "Quality-guided fusion-based co-saliency estimation for image co-segmentation and colocalization," *IEEE Trans. Multimedia*, vol. 20, no. 9, pp. 2466–2477, Sep. 2018.

[49] C. Kanan, M. H. Tong, L. Zhang, and G. W. Cottrell, "SUN: Top-down saliency using natural statistics," *Visual Cognition*, vol. 17, no. 6-7, pp. 979–1003International Journal of Computer Vision, 2009.

[50] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[51] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 779–788.

[52] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results," *Int. J. Comput. Vision*, vol. 88, no. 2, pp. 303–338, 2010.

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

[54] C. Wang, H. Zhang, L. Yang, X. Cao, and H. Xiong, "Multiple semantic matching on augmented n-partite graph for object co-segmentation," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 5825–5839, Dec. 2017.

[55] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 1265–1274.

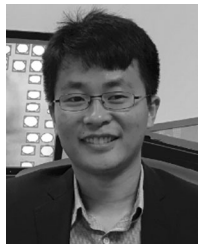
[56] M. Rubinstein, A. Joulin, J. Kopf, and C. Liu, "Unsupervised joint object discovery and segmentation in internet images," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2013, pp. 1939–1946.

[57] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming," Sep. 2013. [Online]. Available: http://cvxr.com/cvx

[58] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 2080–2088.

[59] T. Liu *et al.*, "Learning to detect a salient object," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 353–367, Feb. 2011.

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



Hongkai Yu (Member, IEEE) received the Ph.D. degree in computer science and engineering from the University of South Carolina, Columbia, SC, USA, in 2018. He is an Assistant Professor with the Department of Electrical Engineering and Computer Science, Cleveland State University, Cleveland, OH, USA, since 2020. His research interests include computer vision, machine learning, and deep learning and intelligent transportation system.

Kang Zheng received the B.E. degree in electrical engineering from the Harbin Institute of Technology, Harbin, China, in 2012, and the Ph.D. degree in computer science and engineering from the University of South Carolina, Columbia, SC, USA, in 2019. He is currently a Senior Research Scientist with PAII Inc., Palo Alto, CA, USA His research interest includes computer vision, deep learning and medical image analysis.

Jianwu Fang received the Ph.D. degree in signal and information processing from the University of Chinese Academy of Sciences, Beijing, China, in 2015. He is currently an Associate Professor with the Laboratory of Traffic Vision Safety, School of Electronic and Control Engineering, Chang'an University, Xi'an, China, and is also a Postdoctoral Researcher with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, China. He has authored many papers on top-ranked journals and conferences, such as IEEE-TCYB, IEEE-TIE, IEEE-TCSVT, AAAI, ICRA, etc. His research interests include computer vision and pattern recognition.

Hao Guo received the B.S. degree in computer science and technology and the M.S. degree in signal and information processing from Beijing Jiaotong University, Beijing, China, in 2012 and 2015, respectively. He is currently working toward the Ph.D. degree in computer science with the University of South Carolina, Columbia, SC, USA. His research interests include computer vision and machine learning.

Song Wang (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign (UIUC), Champaign, IL, USA, in 2002. He was a Research Assistant with the Image Formation and Processing Group, Beckman Institute, UIUC, from 1998 to 2002. In 2002, he joined the Department of Computer Science and Engineering, University of South Carolina, Columbia, SC, USA, where he is currently a Professor. His current research interests include computer vision, image processing, and machine learning. He is currently serving as the Publicity/Web Portal Chair of the Technical Committee of Pattern Analysis and Machine Intelligence of the IEEE Computer Society, an Associate Editor for the IEEE TRANSACTION ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, *Pattern Recognition Letters*, and *Electronics Letters.* He is a member of the IEEE Computer Society.