Image Co-saliency Detection and Instance Co-segmentation using Attention Graph Clustering based Graph Convolutional Network

Tengpeng Li, Kaihua Zhang*, Shiwen Shen, Bo Liu, Qingshan Liu, Zhu Li

***Abstract*—Co-Saliency Detection (CSD) is to explore the concurrent patterns and salient objects from a group of relevant images, while Instance Co-Segmentation (ICS) aims to identify and segment out all of these co-salient instances, generating corresponding mask for each instance. To simultaneously tackle these two tasks, we present a novel adaptive graph convolutional network with attention graph clustering (GCAGC) for CSD and ICS, termed as GCAGC-CSD and GCAGC-ICS, respectively. The GCAGC-CSD contains three key model designs: first, we develop a graph convolutional network architecture to extract multi-scale representations to characterize the intra- and inter-image consistency. Second, we propose an attention graph clustering algorithm to distinguish the salient foreground objects from common areas in an unsupervised manner. Third, we present a unified framework with encoder-decoder structure to jointly train and optimize the graph convolutional network, attention graph cluster, and CSD decoder in an end-to-end fashion. Afterwards, we design a salient instance segmentation network for GCAGC-ICS, and combine the outputs of GCAGC-CSD and the instance segmentation branch to obtain instance-aware co-segmentation masks. The proposed GCAGC-CSD and GCAGC-ICS are extensively evaluated on four CSD benchmark datasets (iCoseg, Cosal2015, COCO-SEG and CoSOD3k) and five ICS benchmark datasets (CoSOD3k, COCO-NONVOC, COCO-VOC, VOC12 and SOC), and achieve superior performance over state-of-the-arts on both tasks.**

I. Introduction

Human prefer to pay visual attention on those attractive and interesting regions and objects for future processing [1]. Co-Saliency Detection (CSD) model simulates the human visual system to perceive the scenario, and searches for the co-occurrent and foreground objects in a group of relevant images. CSD has been widely applied to many other tasks to boost the performance of image/video content, such as image/video co-segmentation [2]–[4], imge/video salient object detection [5]–[7], object co-localization [8], [9], and image retrieval [10].

It is inexplicit about the semantical category of the relevant salient objects in CSD problem. Thus, the information of such specific category needs to be inferred by our proposed algorithm from analytical contents of the given image group. Therefore, two key challenges are generally addressed by the CSD algorithm design: (1) extracting valuable image feature representations to robustly describe the image foreground information; and (2) designing a compactness and effective computational framework to formulate and detect the common patterns. The conventional hand-engineered features, such as Gabor filters, color histograms and SIFT descriptors [11] have been widely applied in many CSD models [12]–[14]. However, hand-crafted shallow features are usually not capable of entirely capturing the large variations of common object appearances, and intricate background textures [15]. Recently, researchers significantly improve CSD results to a high level by using deep-learning-based semantical feature representations, and have shown favorable performance [16]–[18]. Nonetheless, these approaches separate the feature extraction from CSD as two distinct steps, and lose the capacity to tailor the image representations towards inferring co-salient regions [19]. The end-to-end frameworks with convolutional neural networks (CNNs) [15], [19], [20] have been designed to tackle this challenge, and shown state-of-the-art performance. Although CNN is able to extract image representations in a data-driven style, it is a sub-optimal solution to model long-range dependencies [21]. The CNNs capture long-range dependencies by progressively stacking convolutional layers on low-resolution features to enlarge the receptive fields. However, the repeated convolutional operations lead to optimization difficulties [21], [22], and make multi-hop dependency modeling [21]. Moreover, it becomes even more challenging for the CNNs to correctly modeling the inter-image non-local dependencies for common salient foregrounds in the image group.

To address the aforementioned challenges, we develop a novel adaptive graph convolutional network with attention graph clustering (GCAGC) for CSD, termed as GCAGC-CSD. We first use a CNN encoder to extract multi-scale and multi-layer feature representations from the image group, and generate combined dense feature node graphs. We then process the dense graphs with the proposed adaptive graph convolutional network (AGCN). Compared with only depending on the progressive behavior of the CNN, the AGCN is able to capture the non-local and long-range correspondence directly by computing the interactions between any two positions of the image group, regardless of their intra- and inter-image

Tengpeng Li, Kaihua Zhang and Qingshan Liu are with Jiangsu Key Laboratory of Big Data Analysis Technology (B-DAT) and Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology (CICAEET), Nanjing University of Information Science and Technology, Nanjing, China. **Corresponding author*: Kaihua Zhang. Email: zhkhua@gmail.com. This work is supported in part by National Major Project of China for New Generation of AI (No. 2018AAA0100400), in part by the Natural Science Foundation of China under Grant nos. 61876088, 61825601, and U20B2065, in part by the 333 High-level Talents Cultivation Project of Jiangsu Province under Grant nos. BRA2020291.

Shiwen Shen and Bo Liu are with JD Digits, Mountain View, California, USA.

Zhu Li is with the Dept of Computer Science & Electrical Engnieering (CSEE), University of Missouri, Kansas City. Zhu Li is supported in part by NSF award 1747751.

Manuscript received April 19, 2005; revised August 26, 2015.

positional distance. The output from AGCN is further refined by an attention graph clustering module (AGCM) through generated co-attention maps. A CNN decoder is deployed in the end of the GCAGC-CSD model to output the predicted co-saliency maps. After achieving the predicted co-saliency maps, we further design an Instance Co-Segmentation (ICS) model, termed as GCAGC-ICS, which uses the Mask R-CNN [23] like network to generate instance segmentation results, and then post-processes the results using the predicted co-saliency maps, generating the desirable ICS results. By simultaneously exploring these two tasks, the predictions of CSD can further boost the performance of ICS by accurately segmenting the common areas in the ICS domain, and the attentive foreground objects and the specific individuals discovered by ICS method can also contribute to finer CSD predictions. We conduct extensive evaluations on four CSD benchmark datasets including iCoseg [24], Cosal2015 [17], COCO-SEG [15] and CoSOD3k [25] and five ICS benchmark datasets including CoSOD3k [25], COCO-NONVOC [26], COCO-VOC [26], VOC12 [26] and SOC [26], and the proposed GCAGC-CSD and GCAGC-ICS achieve favorable performance against a variety of state-of-the-art methods.

This paper is built upon our CVPR work [27] and we significantly extend it in a variety of aspects including conceptual instructions, methodology summary and experimental analysis. First, we add more detailed introductions in the introduction and related work sections. Second, we further propose a salient instance segmentation network branch that is combined with the original GCAGC architecture in [27] to realize ICS task. Third, our original GCAGC model has only been evaluated on three CSD benchmark datasets including iCoseg, Cosal2015 and COCO-SEG. Besides, we conduct more substantial experiments on both the CSD and ICS benchmark datasets including CoSOD3k, COCO-NONVOC, COCO-VOC, VOC12, and SOC.

II. Related Work

Image CSD. Different from salient object detection [29]–[31] or RGB-D salient object detection [32], [33] which only masks the foreground area of the single image, this task highlights both common and salient foreground objects against backgrounds and segments these objects from a group of images. Multiple algorithms have been presented for tackling this task. Approaches can be generally divided into three main strategies including bottom-up methods, fusion-based methods and learning-based methods. Bottom-up methods first score each pixel/sub-region from multiple images, and then combine correspond regions in a bottom-up style. Hand-crafted features [12]–[14], [34]–[36] or deep-learning-based features [16], [17] are widely deployed to measure such sub-regions. Fu *et al.* [12] leverage three visual attentive cues in a cluster-based framework. Liu *et al.* [13] identify background and foreground priors to acquire the intra- and inter-image similarities. Pre-trained CNN and restricted Boltzmann machine are utilized in [16] and [17] to abstract conceptual information for segmenting co-occurrent foreground objects, respectively. In contrast, fusion-based algorithms [2], [37], [38] are developed to explore valuable associations from existing predicted maps generated by several existing saliency or CSD approaches. These methods integrate the explored region proposals by region-wise adaptive fusion [2], adaptive weight fusion [38] or stacked-autoencoder-enabled fusion [37]. Moreover, in the RGB-D CSD task, a bagging-based clustering algorithm is introduced to produce multi-level candidate results for final integration in [39]. Learning-based approaches are the last category of CSD algorithms, and designed to jointly learn common patterns and salient cues automatically from the image group. In [19], an unsupervised CNN framework is proposed to learn the intra-image saliency and cross-image concurrency with two defined graph-based losses, respectively. Zhang *et al.* [18] propose a hierarchical framework in a coarse to fine fashion to detect the co-salient objects with a mask-guided fully CNN. Afterwards, a semantic guided multi-scale feature aggregation model is proposed to capture the concurrent and fine-grained representations in [15]. Ren *et al.* [40] recently propose a deep-learning-based model to combine multi-layers features and design an inter-image propagation mechanism for co-saliency map generations. In [41], a common feature extraction module and a top-down feature fusion module are presented to explore correspondent information. Although many methods have been developed, this field still lacks of research on addressing the limitations of CNN for capturing long-range intra- and inter-image dependencies.

Graph Neural Networks (GNNs). GNNs models [42], [43] are designed to capture graph dependencies through message passing between the nodes of graphs. Different from standard neural network, GNNs retain a state that can acquire informative representations from its neighborhood with arbitrary depth [44]. Convolutional graph neural networks (GCNs) [45]–[48] are a variant of GNNs, and their goal is to produce convolution to graph domain. Algorithms in this branch are usually classified as the spectral-based approaches [45], [46], and the spatial-based approaches [47], [48]. The former ones cooperate with a spectral representation of the graphs; and the latter ones identify the operation directly on graph, and obtain valuable information from groups of spatially connected neighbours. Recently, GNN and GCN have demonstrated favorable results in different computer vision fields, including scene graph generation [49], point clouds classification and segmentation [50], [51], semantic segmentation [52], [53], action recognition [54] and visual reasoning and question answering [55]. More detailed review of GNNs is provided in [44], [56].

Graph Clustering. This task classify the graph nodes into relevant groups. In early years, traditional works [57], [58] develop shallow features for graph clustering. Girvan *et al.* [57] use centrality indices to explore boundaries of different nodes groups. Wang *et al.* [59] propose a modularized non-negative matrix factorization approach to embed the community structure into the graph embedding, and then perform conventional clustering approaches on the embedded features. The limitations of these works are that they only handle partial graph structure or shallow relationships between the content and the structure data [60]. In contrast, deep-learning-based approaches [60], [61] are developed recently to improve graph

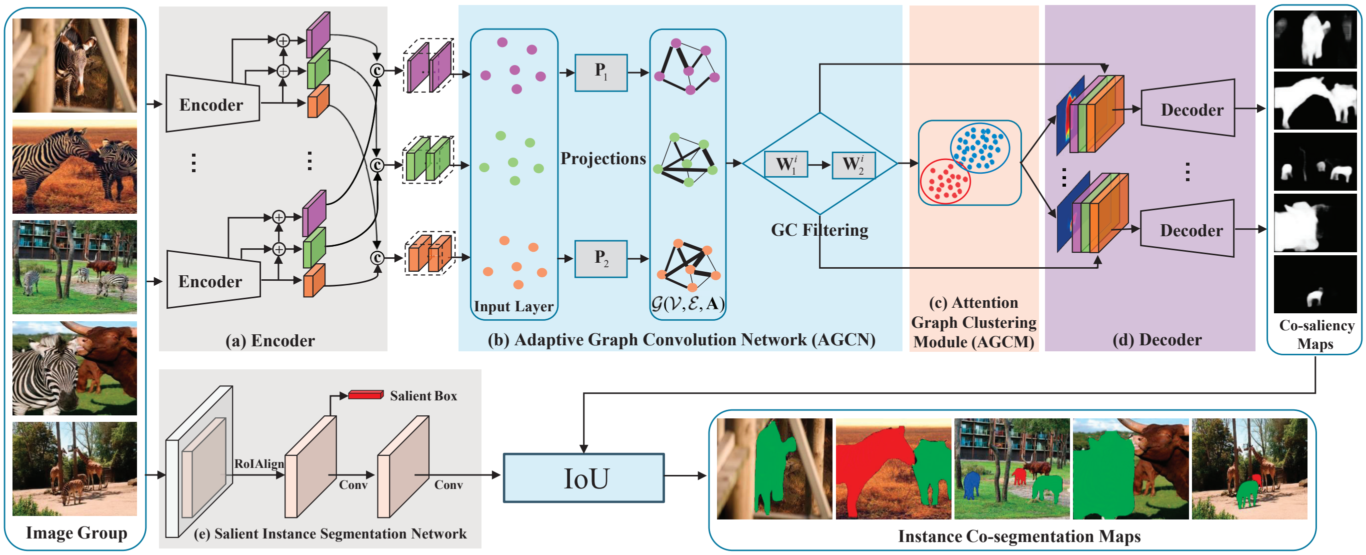
Image Group
Encoder
Encoder
(a) Encoder
Input Layer
P1
P2
Projections
$\mathcal{G}(\mathcal{V},\mathcal{E},\mathbf{A})$
W1
W2
GC Filtering
(b) Adaptive Graph Convolution Network (AGCN)
(c) Attention Graph Clustering Module (AGCM)
Decoder
Decoder
(d) Decoder
Co-saliency Maps
Salient Box
RoIAlign
Conv
Conv
(e) Salient Instance Segmentation Network
IoU
Instance Co-segmentation Maps

Fig. 1: Illustrations of our proposed framework for CSD and ICS. For a group of input images, we first utilize a backbone CNN as encoder (a) to obtain the multi-scale features of each image, and then we introduce the feature pyramid network (FPN) [28] to assemble multi-scale features in a top-down pathway. Next, the lateral output features used as node representations are sent into the AGCN (b). The output features of AGCN through two-layer GCNs are then fed into the AGCM (c), generating a group of co-attention maps. Moreover, the co-attention maps and the output features of AGCN are concatenated and sent into the decoder (d), generating corresponding co-saliency estimations. Finally, the output instance masks of the salient instance segmentation network (e) are further refined with the co-saliency predictions to produce the ICS results. $\oplus$: element-wise addition; ⓒ: concatenation; $\mathcal{G}(\mathcal{V},\mathcal{E},\mathbf{A})$: graph of nodes $\mathcal{V}$, edges $\mathcal{E}$ and adjacency matrix $\mathbf{A}$; $\mathbf{P}_1^k$, $\mathbf{P}_2^k$: learnable projection matrices for graph learning; $\mathbf{W}_1^k$ and $\mathbf{W}_2^k$: learnable weight matrices in the adopted two-layer GCNs.

clustering. Pan *et al.* [61] present an adversarially regularized framework to extract the graph representation to perform graph clustering. Wang *et al.* [60] develop a goal-directed deep learning approach to jointly learn graph embedding and graph clustering together. More comprehensive review of graph clustering can be found in [62].

Instance Segmentation. Instance segmentation can be generally categorized into class-specific and class-agnostic methods [26], [63], [64]. First, the class-specific methods [23], [65], [66] aim at segmenting the instance masks and annotating the specific category according to the training dataset. Among them, detection-based methods have been widely used which depend on a conventional object detector to generate each individual object box and then segment the corresponding instance mask. A typical example is Mask R-CNN [23] that adds an extra pixel-level segmentation branch in parallel with the bounding box detection branch to simultaneously localize and segment individual target. Recently, the class-agnostic methods are becoming more and more popular due to their promising generalization capability to unseen categories in the training set. In [63], a new task of salient instance segmentation has been proposed that leverages a multi-scale saliency refinement network to generate high-quality salient region masks, which are further post-processed using the object proposal boxes and CRF to generate salient instance segmentation results. In [26], an unsupervised framework containing two separated branches including co-peak searching and instance masking is designed for ICS task. In [64], an end-to-end framework is proposed to segment the salient instances regardless of annotating related categories, demonstrating favorable performance against state-of-the-arts.

III. Proposed Approach

A. *Method Overview of GCAGC-CSD*

For a group of N associated images $\mathcal{I} = \{\boldsymbol{I}^n\}_{n=1}^N$, the goal of CSD task is to discriminate the co-occurrent attentive foregrounds from backgrounds, segmenting out the corresponding co-saliency estimations $\mathcal{M} = \{\mathbf{M}^n\}_{n=1}^N$. To achieve this goal, we present a unified GCAGC model to estimate $\mathcal{M}$ in an end-to-end manner.

Figure 1 illustrates the pipeline of our approach, which consists of four key components: (a) Encoder, (b) AGCN, (c) AGCM and (d) Decoder. Specifically, given input $\mathcal{I}$, we first leverage the VGG16 backbone network [67] as the encoder to extract their features by removing the fully-connected layers and softmax layer. Afterwards, we introduce the FPN [28] to integrate the features of pool3, pool4 and pool5 layers, generating three lateral intermediate feature representations $\mathcal{X} = \{\mathbf{X}^k\}_{k=1}^3$ as the multi-scale feature maps of $\mathcal{I}$, Then, for each $\mathbf{X}^k \in \mathcal{X}$, we present a sub-graph $\mathcal{G}^k$ with a learnable structure that is adaptive to our CSD task, which is capable of well capturing the long-range intra- and inter-image similarities while preserving the spatial structure dependencies of the saliency. Meanwhile, the sub-graphs are combined into a multi-graph $\mathcal{G} = \cup_k \mathcal{G}^k$ to deeply explore

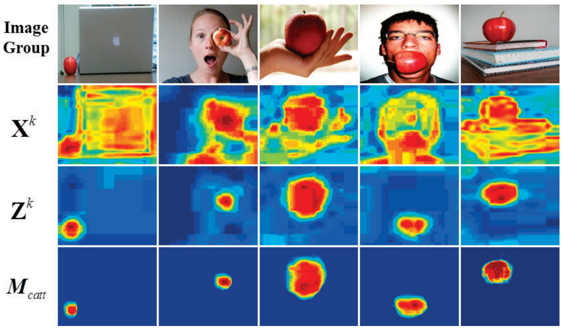

Fig. 2: Illustration of the effect of GC filtering. The GC filtered signal projections $\mathbf{Z}^k$ preserve better spatial consistency of the salient foregrounds than the input graph signals $\mathbf{X}^k$ that highlight more background clutters. Afterwards, the co-attention maps $\boldsymbol{M}_{catt}$ generated by our AGCM in § III-C further remove the interferences of backgrounds existing in $\mathbf{Z}^k$.

multi-scale information for enhanced feature representations. Then, $\mathcal{G}$ is integrated into a simple two-layer GCNs $\mathcal{F}_{gcn}$ [68], generating the projected GC filtered features $\mathcal{F}_{gcn}(\mathcal{X}) = \{\mathcal{F}_{gcn}(\mathbf{X}^k)\}_{k=1}^3$. Recent works [69], [70] show that the GC filtering of GCNs [68] is a Laplacian smoothing process, and hence it makes the salient foreground features of the same category similar, thereby well preserving spatial consistency of the foreground saliency, which facilitates the subsequent intra- and inter-image correspondence. Afterwards, $\mathcal{F}_{gcn}(\mathcal{X})$ are fed into a graph clustering module $\mathcal{F}_{gcm}$, producing a group of co-attention maps $\boldsymbol{M}_{catt}$, which help to further refine the predicted co-salient foregrounds while suppressing the noisy backgrounds. Finally, the concatenated features $\boldsymbol{M}_{catt}$©$\mathcal{F}_{gcn}(\mathcal{X})$ are fed into a decoder layer, producing the finally predicted co-saliency maps.

B. Adaptive Graph Convolution Network

As mentioned above, the goal of designing the AGCN is to optimize features with Laplacian smoothing process [69] which capture long-range intra- and inter-image correspondence while preserve spatial consistency. Numerous graph based works for CSD [3], [19], [26], [71], [72] have been developed to better protect spatial consistency, but they identify intra-saliency detection and inter-image correspondence independently, which cannot fully explore the associations between common patterns and salient foregrounds simultaneously which are necessary to CSD, resulting in sub-optimal performance. In contrast, our AGCN builds a dense graph to take all input image features as the node representations. Meanwhile, each edge of the graph constructs the interactions between any pair-wise nodes regardless of their positional distance, thereby well capturing long-range correspondence. Hence, both intra-saliency detection and inter-image correspondence can be jointly accomplished by feature propagation on the graph under a unified framework without any post-processing, facilitating to segment more accurate co-saliency areas than those individually dealing with each part [3], [19], [26], [71], [72].

Notations of Graph. We build a multi-graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{A}) = \cup_{k=1}^3 \mathcal{G}^k(\mathcal{V}^k, \mathcal{E}^k, \mathbf{A}^k)$ that is consist of three sub-graphs $\mathcal{G}^k$, where node set $\mathcal{V} = \{\mathcal{V}^k\}$, edge set $\mathcal{E} = \{\mathcal{E}^k\}$, adjacent matrix $\mathbf{A} = \sum_k \mathbf{A}^k$, $\mathcal{V}^k = \{v_i^k\}$ denotes the node set of $\mathcal{G}^k$ with node v_i^k, $\mathcal{E}^k = \{e_{ij}^k\}$ denotes its edge set with edge e_{ij}^k, $\mathbf{A}^k$ denotes its adjacent matrix, whose entry $\mathbf{A}^k(i,j)$ denotes the weight of edge e_{ij}^k. $\mathbf{X}^k = [\boldsymbol{x}_1^k, \ldots, \boldsymbol{x}_{Nwh}^k]^\top$ denotes the feature matrix of $\mathcal{G}^k$, where $\boldsymbol{x}_i^k \in \mathbb{R}^{d^k}$ is the features of node v_i^k with dimension d^k.

Adjacency Matrix A. The vanilla GCNs [68] construct a fixed graph without training, which cannot guarantee the optimal performance to a specific task [73]. Recently, adaptive graph learning techniques by learning a dynamic adjacent matrix according to a specific task have been introduced in some works [72]–[74], obtaining promising results. Motivated by these works and the self-attention mechanism in [21], we define a learnable adjacency matrix to learn a task-specific graph structure for sub-graph k, which can be denoted as

$$\mathbf{A}^k = \sigma(\mathbf{X}^k \mathbf{P}_1^k (\mathbf{X}^k \mathbf{P}_2^k)^\top), \tag{1}$$

where $\sigma(x) = \frac{1}{1+e^{-x}}$ denotes the sigmoid function, $\mathbf{P}_1^k, \mathbf{P}_2^k \in \mathbb{R}^{d^k \times r}$ are two learnable projection matrices that reduce the dimension of the node features from d^k to $r < d^k$.

To integrate multiple graphs in GCNs, as in [75], we simply element-wisely add the adjacency matrices of all $\mathcal{G}^k$ to construct the adjacency matrix of $\mathcal{G}$ as

$$\mathbf{A} = \mathbf{A}^1 + \mathbf{A}^2 + \mathbf{A}^3. \tag{2}$$

Graph Convolutional Filtering. We deploy the two-layer GCNs proposed by [68] to perform graph convolutions as

$$\begin{aligned} \mathbf{Z}^k &= \mathcal{F}_{gcn}(\mathbf{X}^k) \\ &= \mathcal{F}_{softmax}(\hat{\mathbf{A}}\mathrm{ReLU}(\mathcal{F}_{gcf}(\hat{\mathbf{A}}, \mathbf{X}^k)\mathbf{W}_1^k)\mathbf{W}_2^k), \end{aligned} \tag{3}$$

where the GC filtering function is defined as [70]

$$\mathcal{F}_{gcf}(\hat{\mathbf{A}}, \mathbf{X}^k) = \hat{\mathbf{A}}\mathbf{X}^k, \tag{4}$$

$\mathbf{W}_1^k \in \mathbb{R}^{d^k \times c_1^k}$, $\mathbf{W}_2^k \in \mathbb{R}^{c_1^k \times c^k}$ denote the learnable weight matrices of two fully-connected layers for feature projections, $\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}$, where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, $\mathbf{A}$ is defined by (2) and $\mathbf{I}$ denotes the identity matrix, $\tilde{\mathbf{D}}(i,i) = \sum_j \tilde{\mathbf{A}}(i,j)$ is the degree matrix of $\tilde{\mathbf{A}}$ that is diagonal. Recent work [70] has proven that the GC filtering $\mathcal{F}_{gcf}$ (4) is low-pass and hence it can ensure smoothed characteristic of the output signal projections $\mathbf{Z}^k$ in the same cluster, so as to well protect the spatial consistency of the salient objects across multiple images as illustrated by Figure 2. However, some intra-consistency but non-salient regions have also been highlighted. To tackle this problem, in the following section, we will present an attention graph clustering technique to further optimize $\mathbf{Z}^k$ to retain co-attention regions while removing non-salient common areas.

C. Attention Graph Clustering Module

Figure 3 introduces the schematic diagram of our AGCM $\mathcal{F}_{gcm}$. Specifically, given the GC filtering projections $\mathbf{Z}^k \in \mathbb{R}^{Nwh \times c^k}, k = 1, 2, 3$ in (3), we obtain a multi-scale feature matrix by concatenating them as $\mathbf{Z} = [\mathbf{Z}^1, \mathbf{Z}^2, \mathbf{Z}^3] =$

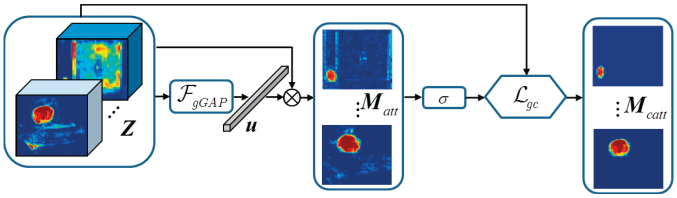

Fig. 3: The schematic diagram of our AGCM $\mathcal{F}_{gcm}$. Please refer to the text part for details.

$[\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{Nwh}]^\top \in \mathbb{R}^{Nwh \times d}$, where the multi-scale node features $\boldsymbol{z}_i \in \mathbb{R}^d$, $d = \sum_k c^k$. Next, we reshape $\mathbf{Z}$ to tensor $\boldsymbol{Z} \in \mathbb{R}^{N \times w \times h \times d}$ as input of $\mathcal{F}_{gcm}$. Then, we define a group global average pooling (gGAP) function $\mathcal{F}_{gGAP}$ as

$$\boldsymbol{u} = \mathcal{F}_{gGAP}(\mathbf{Z}) = \frac{1}{Nwh} \sum_{n,i,j} \boldsymbol{Z}(n, i, j, :), \tag{5}$$

which generates a global statistic feature $\boldsymbol{u} \in \mathbb{R}^d$ as the multi-scale semantic saliency representation that discovers the global useful group-wise context information. Afterwards, we correlate $\boldsymbol{u}$ and $\mathbf{Z}$ to produce a group of foreground maps that can fully highlight the intra-saliency:

$$\boldsymbol{M}_{att} = \boldsymbol{u} \otimes \mathbf{Z}, \tag{6}$$

where $\boldsymbol{M}_{att} \in \mathbb{R}^{N \times w \times h}$, $\otimes$ denotes correlation operator. Then, we use sigmoid function σ to re-scale the values of $\boldsymbol{M}_{att}$ to $[0, 1]$ as

$$\boldsymbol{W} = \sigma(\boldsymbol{M}_{att}). \tag{7}$$

From Figure 3, we can find that $\boldsymbol{M}_{att}$ encodes intra-saliency that preserves spatial consistency, but some noisy non-co-salient foregrounds have also been highlighted. To alleviate this issue, an attention graph clustering algorithm is exploited to further optimize the attention maps that can be capable of differentiating the common objects from foregrounds. Inspired by the weighted kernel k-means approach in [76], we define the objective function of AGCM as

$$\mathcal{L}_{gc} = \sum_{\boldsymbol{z}_i \in \pi_f} w_i \|\boldsymbol{z}_i - \boldsymbol{m}_f\|^2 + \sum_{\boldsymbol{z}_i \in \pi_b} w_i \|\boldsymbol{z}_i - \boldsymbol{m}_b\|^2, \tag{8}$$

where π_f and π_b represent the clusters of foregrounds and backgrounds, respectively, $\boldsymbol{m}_f = \frac{\sum_{\boldsymbol{z}_i \in \pi_f} \boldsymbol{z}_i w_i}{\sum_{\boldsymbol{z}_i \in \pi_f} w_i}$ and similar for $\boldsymbol{m}_b$, w_i denotes the i-th element of $\boldsymbol{W}$ in (7).

Following [76], we can readily show that the minimization of the objective $\mathcal{L}_{gc}$ in (8) is equivalent to

$$\min_{\mathbf{Y}} \{\mathcal{L}_{gc} = -\text{trace}(\mathbf{Y}^\top \mathbf{K} \mathbf{Y})\}, \tag{9}$$

where $\mathbf{K} = \mathbf{D}^{\frac{1}{2}} \mathbf{Z} \mathbf{Z}^\top \mathbf{D}^{\frac{1}{2}}$, $\mathbf{D} = \text{diag}(w_1, \ldots, w_{Nwh})$, $\mathbf{Y} \in \mathbb{R}^{Nwh \times 2}$ satisfies $\mathbf{Y}^\top \mathbf{Y} = \mathbf{I}$.

Let $\boldsymbol{y} \in \{0, 1\}^{Nwh}$ denote the indictor vector of the clusters, and $\boldsymbol{y}(i) = 1$ if $i \in \pi_f$, else, $\boldsymbol{y}(i) = 0$. We choose $\mathbf{Y} = [\boldsymbol{y}/\sqrt{|\pi_f|}, (\mathbf{1} - \boldsymbol{y})/\sqrt{|\pi_b|}]$ that satisfies $\mathbf{Y}^\top \mathbf{Y} = \mathbf{I}$ and put it into (9), yielding the loss function of our AGCM

$$\mathcal{L}_{gc} = -\left(\frac{\boldsymbol{y}^\top \mathbf{K} \boldsymbol{y}}{\boldsymbol{y}^\top \boldsymbol{y}} + \frac{(\mathbf{1} - \boldsymbol{y})^\top \mathbf{K} (\mathbf{1} - \boldsymbol{y})}{(\mathbf{1} - \boldsymbol{y})^\top (\mathbf{1} - \boldsymbol{y})} \right). \tag{10}$$

Now, we show the relationship between the above loss $\mathcal{L}_{gc}$ and graph clustering. We first construct the graph of GC as $\mathcal{G}_{gc}(\mathcal{V}_{gc}, \mathcal{E}_{gc}, \mathbf{K})$, which is made up of node set $\mathcal{V}_{gc} = \mathcal{V}_f \cup \mathcal{V}_b$, where $\mathcal{V}_f$ is the set of foreground nodes and $\mathcal{V}_b$ is the set of background nodes, $\mathcal{E}_{gc}$ denotes the edge set such that the weight of edge between nodes i and j is equal to $\mathbf{K}(i, j)$, where $\mathbf{K}$ is its adjacency matrix defined in (9). Let us denote $\text{links}(\mathcal{V}_l, \mathcal{V}_l) = \sum_{i \in \mathcal{V}_l, j \in \mathcal{V}_l} \mathbf{K}(i, j), l = f, b$, then, it is easy to show that minimizing $\mathcal{L}_{gc}$ (10) is equivalent to maximizing the ratio association objective [77] for graph clustering task

$$\max \left\{ \sum_{l=f,g} \frac{\text{links}(\mathcal{V}_l, \mathcal{V}_l)}{|\mathcal{V}_l|} \right\}. \tag{11}$$

where $|\mathcal{V}_l|$ denotes the cardinality of set $\mathcal{V}_l$.

Directly optimizing $\mathcal{L}_{gc}$ (10) yields its continuous relaxed solution $\hat{\boldsymbol{y}}$. Then, we reshape $\hat{\boldsymbol{y}}$ into a group of N co-attention maps $\boldsymbol{M}_{catt} \in \mathbb{R}^{N \times w \times h}$. Finally, the learned co-attention maps $\boldsymbol{M}_{catt}$ and the input features $\mathbf{Z} \in \mathbb{R}^{N \times w \times h \times d}$ of the AGCM are concatenated, yielding the enhanced features $\boldsymbol{F} \in \mathbb{R}^{N \times w \times h \times (d+1)}$:

$$\boldsymbol{F} = \boldsymbol{M}_{catt} \copyright \mathbf{Z}, \tag{12}$$

where © denotes concatenation operator, which serves as the input of the following decoder network.

D. Decoder Network

In general, our decoder network has an up-sampling module that contains a 3×3 convolutional layer to decrease feature channels, a ReLU layer and a deconvolutional layer with stride $= 2$ to increase resolution. Then, we repeat this module three times until reaching the finest resolution for accurate co-saliency map estimation, following a 1×1 convolutional layer and a sigmoid layer to produce a group of co-saliency map estimations.

Given the features $\boldsymbol{F}$ computed by (12) as input, the decoder network generates a group of co-saliency maps $\mathcal{M} = \{\mathbf{M}^n \in \mathbb{R}^{w \times h}\}_{n=1}^N$. We then leverage a weighted cross-entropy loss for pixel-wise classification

$$\begin{aligned} \mathcal{L}_{cls} = -\frac{1}{P \times N} \sum_{n=1}^{N} \sum_{i=1}^{P} \{ & \rho^n \mathbf{M}^n(i) \log(\mathbf{M}_{gt}^n(i)) \\ & -(1 - \rho^n)(1 - \mathbf{M}^n(i)) \log(1 - \mathbf{M}_{gt}^n(i))\}, \end{aligned} \tag{13}$$

where $\mathbf{M}_{gt}^n$ denotes the ground-truth mask of image $\boldsymbol{I}^n \in \mathcal{I}$, P denotes the pixel number of image $\boldsymbol{I}^n$ and ρ^n denotes the ratio of all positive pixels over all pixels in image $\boldsymbol{I}^n$.

All the network parameters are jointly learned by minimizing the following multi-task loss function

$$\mathcal{L}_{co} = \mathcal{L}_{cls} + \lambda \mathcal{L}_{gc}, \tag{14}$$

where $\mathcal{L}_{gc}$ is the attention graph clustering loss defined by (10), $\lambda > 0$ is a trade-off parameter. We train our network by minimizing $\mathcal{L}$ in an end-to-end manner, and the learned GCAGC model is directly applied to processing input image group, predicting the corresponding co-saliency maps without any post-processing.

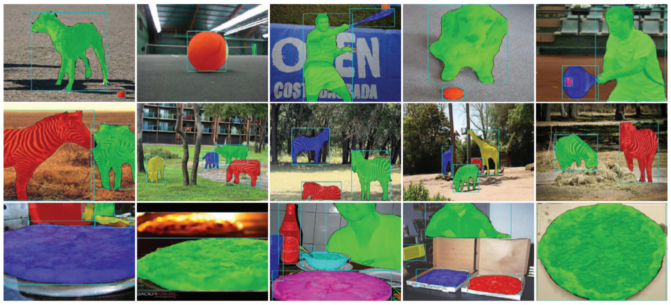

Fig. 4: Visual examples of the salient instance segmentation results by our method, where images in the first, the second and the third rows are selected from *table tennis*, *zebra* and *pizza* group of CoSOD3k dataset, respectively.

E. Extension to ICS

As illustrated by Figure 1, the proposed GCAGC-CSD framework can be readily extended to ICS task by adding a salient instance segmentation network branch. The recently proposed Mask R-CNN [23] provides a strong baseline to instance segmentation. Motivated by this work, we design the salient instance segmentation network, which is integrated into the GCAGC-CSD framework to perform ICS task.

Backbone. We adopt the ResNet-50 backbone [22] pre-trained for the image classification task as the feature extractor of our instance segmentation model. Following feature pyramid network (FPN) [28], we first extract the features from conv2 to conv5 layers and each layer follows a 1×1 convolutional layer to ensure the same channel dimension. Afterwards, a reverse architecture is constructed to sum the up-sampled high-level features and the shallow features in a top-down pathway progressively.

Salient Instance Detector and Segmentation. Similar to the two-stage processing in Mask R-CNN, our network has an identical region proposal network (RPN) in the first stage and a parallel foreground bounding box classification and regression in the second stage. Besides, the network outputs a binary mask for instance prediction. Thus, the loss of our instance segmentation network branch is defined as

$$\mathcal{L}_{ins} = \mathcal{L}^{f}_{cls} + \mathcal{L}_{box} + \mathcal{L}^{f}_{seg}, \tag{15}$$

where $\mathcal{L}^{f}_{cls}$, $\mathcal{L}_{box}$, and $\mathcal{L}^{f}_{seg}$ represent the binary foreground classification loss, the bounding box regression loss and the binary foreground segmentation loss, respectively. Given a group of images $\mathcal{I} = \{\boldsymbol{I}^n\}_{n=1}^{N}$, we obtain corresponding salient instance segmentation results $\mathcal{S} = \{\mathbf{S}^n\}_{n=1}^{N}$, where $\mathbf{S}^n = \{\mathbf{IM}^n_m\}_{m=1}^{M}$ denotes all M instance masks in a single image $\boldsymbol{I}^n$.

Figure 4 shows some visualized salient instance segmentation results of different image groups, which exhibit promising performance of the proposed instance segmentation model even in some challenging scenes with severe background clutters.

Generating ICS Results. Finally, we combine salient instance segmentation mask $\mathbf{S}^n = \{\mathbf{IM}^n_m\}_{m=1}^{M}$ with each relevant co-saliency estimation $\mathbf{M}^n$ generated by GCAGC-CSD to achieve the ICS results $\mathcal{M}^n_{ics} = \{\mathbf{1}^n_m \odot \mathbf{IM}^n_m\}_{m=1}^{M}$, where $\odot$ denotes element-wise product and $\mathbf{1}^n_m$ is a binary indicator matrix for instance m of image $\boldsymbol{I}^n$, whose (i,j)-th element is defined as

$$\mathbf{1}^n_m(i,j) = \begin{cases} 1, IoU(\mathbf{IM}^n_m, \mathbf{M}^n) > 0.7 \\ 0, else, \end{cases} \tag{16}$$

where $IoU(A,B) = \frac{|A \cap B|}{|A \cup B|}$ is an interaction-over-union function that is widely used in object detection [23].

IV. Experimental Results

A. Implementation Details

The training of our GCAGC-CSD model includes two stages:

Stage 1. We select the VGG16 network [67] pre-trained on the ImageNet classification task [78] as our backbone network for fair comparisons. Following the input settings in [15], [20], we randomly choose $N = 5$ images as one group from one category and then select a mini-batch groups from all categories in the COCO dataset [79], which are sent into the network at the same time during training. All the images are resized to the same size of 224×224 for easy processing. The model is optimized by the Adam algorithm [80] with a weight decay of 5e-4 and an initial learning rate of 1e-4 which is reduced by a half every $25,000$ iterations. This training process converges until $100,000$ iterations.

Stage 2. We further fine-tune our model using MSRA-B dataset [81] to better capture the salient objects. The training iterations =$10,000$ and all other parameters are the same as those in **Stage 1**. Especially in training procedure, we extend the number of single salient image to $N = 5$ as a group leveraging affine transformation, horizontal flipping and left-right flipping to match the size of input group. During testing, we divide all images into several mini-groups to generate the final co-saliency maps.

Our GCAGC-ICS model is fine-tuned on the pre-trained Mask R-CNN model [23], where the classification branch is simplified to classify two categories of foreground and background. We consider the ROI as positive when the IoU value of the predicted box and the ground-truth box is larger than 0.5, and is negative otherwise. The training dataset is selected from [63] which contains $10,000$ high-quality salient instance annotations.

The whole framework is implemented in PyTorch with a RTX 2080Ti GPU for acceleration.

B. Datasets and Evaluation Metrics

For CSD, we conduct comprehensive evaluations on four popular datasets including iCoseg [24], Cosal2015 [17], COCO-SEG [15] and CoSOD3k [25]. Among them, iCoseg is the most popular dataset which contain 643 images from 38 groups and its co-attention objects in one image group have shared appearance, common backgrounds and similar conceptual characteristics, but have many challenging factors like various pose or color difference. Cosal2015 is a larger dataset which is proposed to satisfy the requirements of deep-learning technique, which contains 50 semantic categories of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2021.3054526, IEEE Transactions on Multimedia

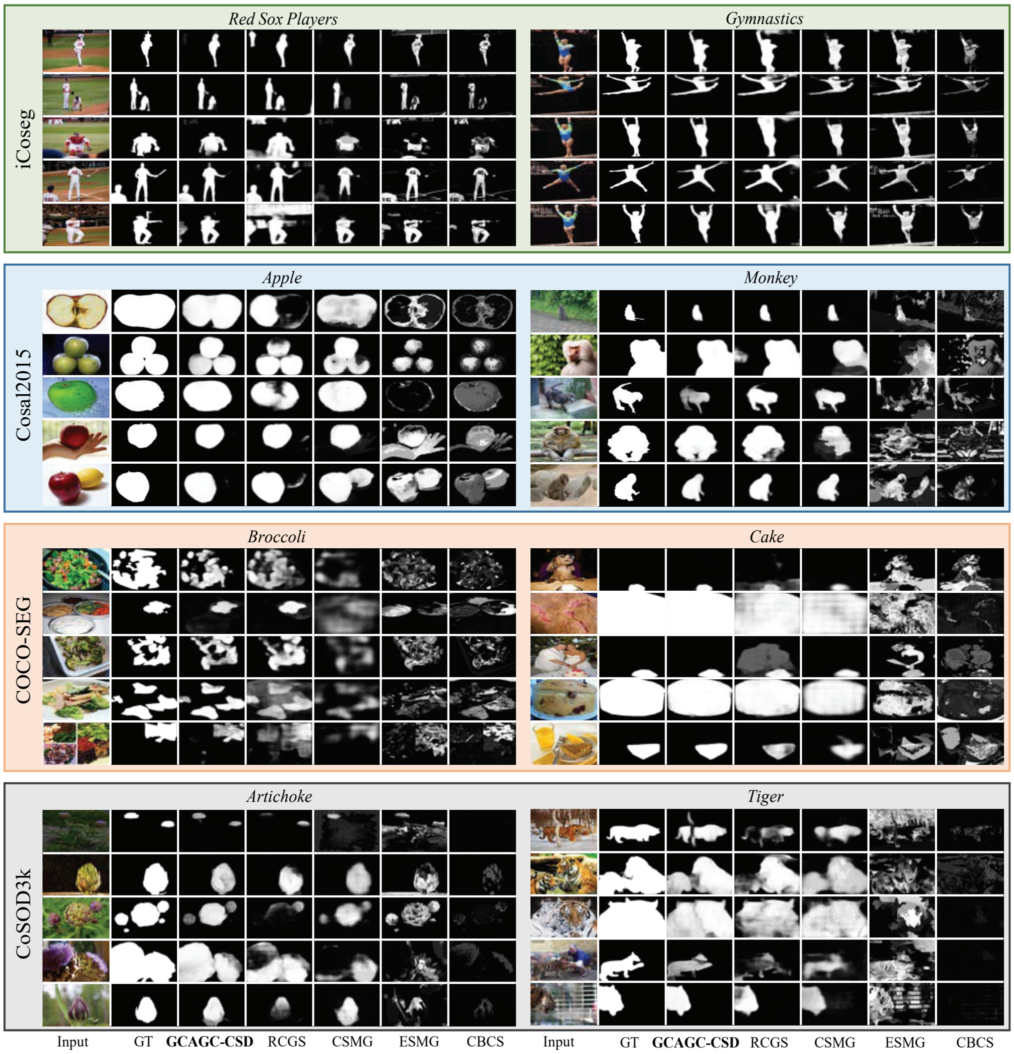

Fig. 5: Visual comparisons of our GCAGC-CSD compared with other state-of-the-arts, including CBCS [12], ESMG [35], CSMG [18] and RCGS [15].

$2,015$ high-quality images, and each group suffers from various negative challenges such as complex scenarios, occlusion problem, appearance variance and noisy backgrounds, *etc*. All these interferences increase the difficulty of detecting accurate co-saliency areas.

Then, to meet the urgent requirement of large-scale training and testing set for deep-learning-based CSD methods, COCO-SEG chosen from the COCO2017 dataset [79] has been proposed, of which $200,000$ and $8,000$ images are for training and testing from all 78 categories, respectively. Recently, CoSOD3k [25] is a large-scale public dataset which owns totally $3,316$ images with 160 groups, including diverse complex scenes with extra bounding-box, instance-level annotations that can be extended to other computer vision tasks. We compare our GCAGC-CSD method with existing state-of-the-art algorithms in terms of 6 metrics including the precision-recall (PR) curve [82], the receive operator characteristic (ROC) curve [82], S-measure score S_α [83], F-measure score F_β [82], E-measure score E_ξ [84] and Mean Absolute Error M [15].

For ICS, we conduct extensive experiments on one instance co-saliency dataset CoSOD3k [25] and four ICS datasets including COCO-VOC, COCO-NONVOC, VOC12 and SOC released from [26]. We adopt the mean average precision (mAP) [85] as the metric to compare the performance of our GCAGC-ICS with other state-of-the-art methods. Specifically, the IoU thresholds of mAP are set to 0.25 and 0.5 following [26], denoted as mAP$^r_{0.25}$ and mAP$^r_{0.5}$, respectively.

C. Comparisons of CSD

We compare our GCAGC-CSD with 12 state-of-the-art CSD methods including CBCS [12], CSHS [13], ESMG [35], SACS [38], CODR [14], CODW [17], DIM [87], UMLF [86], UCSG [19], RCGS [15], CSMG [18] and GICD [88]. Note that we directly leverage public results released by authors or reproduce experimental results by the available source code of each compared method for fair comparison.

TABLE I: Statistic comparisons of our GCAGC-CSD with other state-of-the-arts. **Red** and **blue** bold fonts indicate the best and second best performance, respectively.

	Metric	CBCS [12]	ESMG [35]	CSHS [13]	SACS [38]	CODR [14]	UMLF [86]	DIM [87]	CODW [17]	UCSG [19]	RCGS [15]	CSMG [18]	GICD [88]	**GCAGC -CSD**
iCoseg	$S_\alpha \uparrow$	0.658	0.728	0.750	0.752	0.815	0.703	0.758	0.750	0.820	0.784	0.821	**0.832**	**0.821**
	$F_\beta \uparrow$	0.705	0.685	0.765	0.770	0.823	0.761	0.797	0.782	0.832	0.718	**0.850**	0.835	**0.837**
	$E_\xi \uparrow$	0.797	0.784	0.841	0.817	0.889	0.827	0.864	0.832	0.864	0.818	**0.889**	0.887	**0.897**
	$M \downarrow$	0.172	0.157	0.179	0.154	0.114	0.226	0.179	0.184	0.122	0.098	0.106	**0.080**	**0.078**
Cosal2015	$S_\alpha \uparrow$	0.544	0.552	0.592	0.694	0.689	0.662	0.592	0.648	0.751	0.797	0.774	**0.842**	**0.823**
	$F_\beta \uparrow$	0.532	0.476	0.564	0.650	0.634	0.690	0.580	0.667	0.740	0.779	0.784	**0.829**	**0.831**
	$E_\xi \uparrow$	0.656	0.640	0.685	0.749	0.749	0.769	0.695	0.752	0.805	0.855	0.842	**0.881**	**0.890**
	$M \downarrow$	0.233	0.247	0.313	0.194	0.204	0.271	0.312	0.274	0.160	0.099	0.130	**0.071**	**0.089**
COCO-SEG	$S_\alpha \uparrow$	0.474	0.496	-	0.523	-	0.485	0.457	-	-	**0.721**	0.658	-	**0.739**
	$F_\beta \uparrow$	0.264	0.281	-	0.373	-	0.421	0.271	-	-	**0.645**	0.569	-	**0.666**
	$E_\xi \uparrow$	0.593	0.618	-	0.653	-	0.677	0.625	-	-	**0.809**	0.768	-	**0.823**
	$M \downarrow$	0.226	0.210	-	0.310	-	0.389	0.373	-	-	**0.115**	0.127	-	**0.095**
CoSOD3k	$S_\alpha \uparrow$	0.528	0.532	0.563	-	0.630	0.632	0.559	-	-	0.736	0.711	**0.776**	**0.759**
	$F_\beta \uparrow$	0.466	0.418	0.484	-	0.530	0.639	0.495	-	-	0.682	0.709	**0.723**	**0.730**
	$E_\xi \uparrow$	0.637	0.635	0.656	-	0.700	0.758	0.662	-	-	0.800	0.804	**0.820**	**0.823**
	$M \downarrow$	0.228	0.239	0.309	-	0.229	0.285	0.327	-	-	0.124	0.157	**0.095**	**0.092**

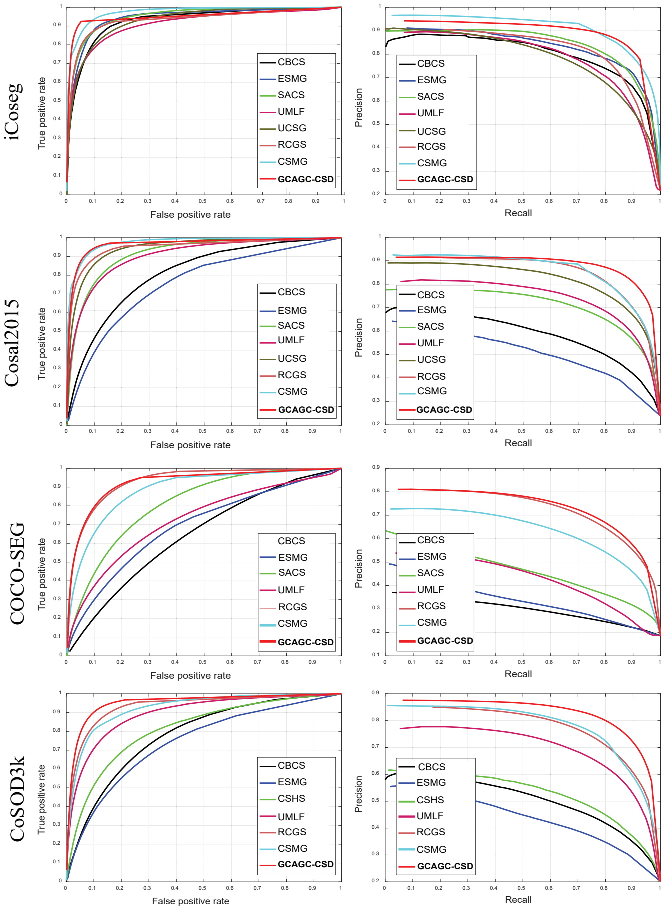

Fig. 6: Comparisons with state-of-the-art methods in terms of PR and ROC curves on four CSD benchmark datasets.

Qualitative Results. Figure 5 shows some visual comparison results with 4 state-of-the-art methods including CBCS [12], ESMG [35], CSMG [18] and RCGS [15]. Our GCAGC-CSD can achieve better co-saliency results than the other methods when the co-salient targets suffer from significant appearance variations, strong semantic interference and complex background clutters. In Figure 5, the top two groups of images are selected from iCoseg. Among them, for the group of *Red Sox Players*, the numerous audience appear in the background have the same category with those baseball players in the foreground, increasing the difficulty of correctly discriminate those people. However, our GCAGC-CSD can accurately detect the co-salient players due to its two-steps filtering processing from GC filtering to graph clustering that can well preserve spatial consistency while effectively reducing noisy backgrounds. In contrast, the other compared approaches represent unsatisfying segmentation results that possess either background clutters (see the left middle rows of RCGS, ESMG, CBCS) or the misleading intra-salient objects including non-co-salient regions (see the left first row of RCGS, the fourth rows of ESMG and CBCS). The predictions of image groups (*Apple* and *Monkey*) in the second row are chosen from Cosal2015 dataset. The *Apple* group should overcome the interferences of other foreground semantic categories such as hand and lemon while the *Monkey* group suffers from complex noisy backgrounds. It is obvious that our GCAGC-CSD can output promising consistent co-saliency maps than the other algorithms (see the left first row of RCGS and CSMG in *Apple* group, the right two columns of ESMG and CBCS in *Monkey* group). The two groups in third row are chosen from COCO-SEG, which is consist of a diverse challenging images with targets suffering from the interferences of multiple semantical categories and complex background clutters. Notwithstanding, our GCAGC-CSD can accurately identify the co-salient targets even when they suffer from extremely complex background clutters (see the *Broccoli* group). The *Artichoke* and *Tiger* groups in the bottom row are chosen from CoSOD3k, which contain various class-agnostic objects with diverse appearances and multiple instances, increasing the difficulty of segmentation. While our GCAGC-CSD can segment out the correct targets regardless of the appearance changes (see the *Artichoke* group) and effectively detect the multiple co-saliency objects (see the *Tiger* group).

The experimental results show that our GCAGC-CSD can achieve favorable performance against various challenging factors, validating the effectiveness of our GCAGC-CSD model that can adapt well to a variety of complicate scenarios.

Quantitative Results. Figure 6 shows the PR and the ROC

TABLE II: Ablative studies of our model on iCoseg and Cosal2015. Here -N, -M, -P denote our GCAGC-CSD in absence of AGCN, AGCM and the projection matrices **P** in (1), respectively. **Red** bold font indicates the best performance.

Datasets		-N	-M	-P	GCAGC-CSD
iCoseg	$S_\alpha \uparrow$	0.818	0.819	0.812	**0.821**
	$F_\beta \uparrow$	0.826	0.822	0.827	**0.837**
	$E_\xi \uparrow$	0.891	0.880	0.883	**0.897**
	$M \downarrow$	0.085	0.081	0.081	**0.078**
Cosal2015	$S_\alpha \uparrow$	0.816	0.814	0.816	**0.823**
	$F_\beta \uparrow$	0.808	0.823	0.821	**0.831**
	$E_\xi \uparrow$	0.856	0.887	0.882	**0.890**
	$M \downarrow$	0.099	0.090	0.096	**0.089**

curves of all compared methods on four benchmark datasets. We can observe that our GCAGC-CSD outperforms the other state-of-the-art methods on four datasets. Especially, all the curves on the largest and most challenging CoSOD3k and COCO-SEG are much higher than the other methods. Meanwhile, Table I lists the statistic analysis, among which the RCGS is a representative end-to-end deep-learning-based method that achieves state-of-the-art performance on both Cosal2015 and COCO-SEG with the F_β of 0.779 and 0.645, respectively. Our GCAGC-CSD achieves the best F_β of 0.831 and 0.666 on Cosal2015 and COCO-SEG, respectively, outperforming the CSMG by 4.7% on Cosal2015 and RCGS by 2.1% on COCO-SEG. All the qualitative results further demonstrate the effectiveness of jointly learning the GCGAC-CSD model that is essential to accurate CSD.

Ablative Studies. Here, we conduct ablative studies to validate the effectiveness of the proposed two modules (AGCN and AGCM) and the adaptive graph learning strategy in the AGCN. Table II lists the corresponding quantitative statistic results in terms of S_α, F_β, E_ξ and M.

First, without AGCN, the GCAGC-CSD-N presents significant performance drop on Cosal2015 in terms of all metrics, especially for both F_β and E_ξ, where the former reduces from 0.831 to 0.808 by 2.3% and the latter reduces from 0.890 to 0.856 by 3.4%. Besides, the performance of GCAGC-CSD-N on iCoseg also shows obvious degradation in terms of all metrics.

Second, without AGCM, the GCAGC-CSD-M suffers from significant performance drop in terms of all metrics on both datasets, especially for F_β and E_ξ on iCoseg, where the F_β and the E_ξ reduce from 0.837 to 0.810 by 2.7% and from 0.897 to 0.877 by 2.0%, respectively. The results demonstrate the effectiveness of the proposed AGCM that can well differentiate the common objects from all salient foregrounds to further boost the performance.

Finally, without adaptive graph learning in AGCN, all metrics in GCAGC-CSD-P have significant decline on both datasets, further showing the superiority of proposed AGCN to learn an adaptive graph structure tailored to the CSD task compared with the fixed graph design in the vanilla GCNs [68].

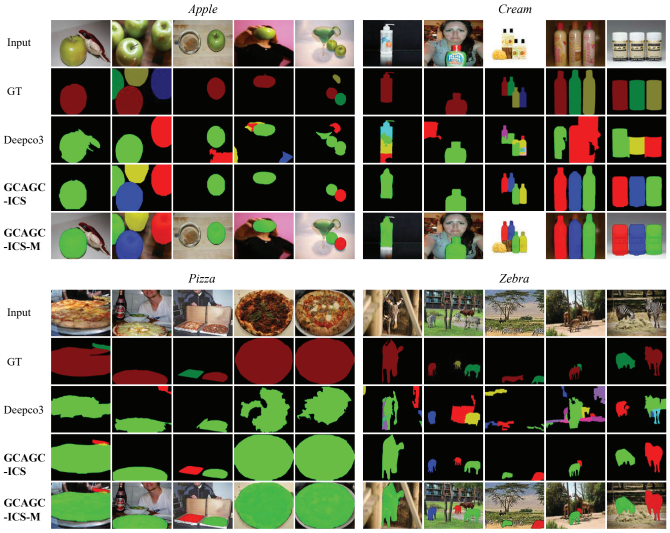

Fig. 7: Visual comparisons of the ICS results with state-of-the-art Deepco3 [26] on CoSOD3k dataset, where **GCAGC-ICS-M** indicates that the ICS results of GCAGC-ICS are masked on the origin images for better visualization.

D. Comparisons of ICS

1) Results on CoSOD3k: We compare the results of GCAGC-ICS with the state-of-the-art method Deepco3 [26], which is the largest and most challenging ICS benchmark dataset by now.

Qualitative Results. Figure 7 visualizes the ICS results of GCAGC-ICS and Deepco3. ICS task needs to deal with diverse issues such as the instance occlusions, background clutters and semantical interferences. For example, the second image of *Apple* group and the third image of *Cream* group encounter the issue of instance occlusion from intra-class objects. Figure 7 shows that our method can faithfully segment out the individual apples and creams regardless of heavy occlusion, while Deepco3 does not work well in these scenes due to its poor ability of discriminating and detecting instances.

Quantitative Results. Table III lists some statistic comparisons on 13 super-classes of CoSOD3k. We can observe that our **overall** scores of metirc $\mathrm{mAP}^r_{0.25}$ and $\mathrm{mAP}^r_{0.5}$ are 0.567 and 0.512, outperforming the Deepco3 by 0.8% and 18.4%, respectively. Specifically, our scores of $\mathrm{mAP}^r_{0.5}$ are 0.632, 0.646 and 0.552 in three sub-classes of *Anim.*, *Ball* and *Cosm.*, which are 21.3%, 36.0% and 16.3% extremely higher than the scores of Deepco3 0.419, 0.286 and 0.389 by a large margin, demonstrating the outstanding performance of GCAGC-ICS.

2) Results on other ICS benchmarks: To further show the superior performance of our GACGC-ICS, we additionally conduct extensive comparisons of our method with 9 state-of-the-arts including CLRW [8], UODL [89], DDT [90], DDT+ [91], DFF [92], NLDF [93], C3SNet [94], PRM [66] and Deepco3 [26] on four public datasets including COCO-VOC, COCO-NONVOC, VOC12 and SOC. For fairness, we compare all existing or reproduced results based on the same evaluation codes released from [26]. In Table IV, the statistic results of GCAGC-ICS show that our GACGC-ICS consis-

TABLE III: Quantitative evaluations of Deepco3, SIS and GCAGC-ICS on 13 super-class of CoSOD3k in terms of $\text{mAP}^r_{0.25}$ and $\text{mAP}^r_{0.5}$ metrics, including Animal (*Anim.*), Cosmetic (*Cosm.*), Electronic (*Elec.*), Instrument (*Inst.*), Kitchenware (*Kitc.*), Necessary (*Nece.*), Others (*Othe.*), Traffic (*Traf.*) and Vegetables (*Vege.*). "***Overall***" means the score on the whole dataset. **Red** bold font indicates the best performance.

Metric	Methods	*Anim.*	*Ball*	*Cosm.*	*Elec.*	*Food*	*Fruit*	*Inst.*	*Kitch.*	*Nece.*	*Othe.*	*Tool*	*Traf.*	*Vege.*	***Overall***
$\text{mAP}^r_{0.25}$	Deepco3 [26]	0.678	0.358	**0.693**	0.576	0.501	0.517	**0.530**	0.606	0.465	**0.407**	**0.418**	0.625	**0.604**	0.559
	SIS	0.556	0.393	0.442	0.418	0.436	0.500	0.169	0.442	0.464	0.271	0.207	0.513	0.214	0.419
	GCAGC-ICS	**0.702**	**0.659**	0.585	**0.628**	**0.708**	**0.593**	0.290	**0.616**	**0.509**	0.374	0.399	**0.671**	0.482	**0.567**
$\text{mAP}^r_{0.5}$	Deepco3 [26]	0.419	0.286	0.389	0.384	0.291	0.340	**0.277**	0.386	0.238	0.175	0.203	0.334	**0.403**	0.328
	SIS	0.517	0.378	0.402	0.394	0.400	0.478	0.106	0.417	0.448	0.233	0.185	0.468	0.197	0.385
	GCAGC-ICS	**0.632**	**0.646**	**0.552**	**0.607**	**0.658**	**0.553**	0.214	**0.545**	**0.487**	**0.321**	**0.346**	**0.600**	0.388	**0.512**

TABLE IV: Statistic comparisons of our GCAGC-ICS, SIS with other state-of-the-arts on four ICS datasets. **Red** and **blue** bold fonts indicate the best and second best performance, respectively.

Datasets	Metric	CLRW	UODL	DDT	DDT+	DFF	NLDF	C2S-Net	PRM	Deepco3	SIS	**GCAGC**
		[8]	[89]	[90]	[91]	[92]	[93]	[94]	[66]	[26]		**-ICS**
COCO-VOC	$\text{mAP}^r_{0.25}$	33.3	9.6	31.4	31.7	30.8	39.1	39.6	44.9	52.6	**90.5**	**91.3**
	$\text{mAP}^r_{0.5}$	13.7	2.2	10.1	10.6	11.6	18.2	13.4	14.6	21.1	**85.5**	**86.9**
COCO-NONVOC	$\text{mAP}^r_{0.25}$	24.6	8.5	25.7	26.0	22.6	23.9	25.1	-	35.3	**79.0**	**79.4**
	$\text{mAP}^r_{0.5}$	10.7	1.8	9.7	10.1	7.3	8.5	7.6	-	12.3	**71.4**	**76.8**
VOC12	$\text{mAP}^r_{0.25}$	29.2	9.4	30.7	33.6	27.7	34.3	30.1	45.3	45.6	**78.3**	**80.4**
	$\text{mAP}^r_{0.5}$	10.5	2.0	8.8	9.4	13.7	12.7	10.7	14.8	16.7	**73.9**	**75.4**
SOC	$\text{mAP}^r_{0.25}$	34.9	11.0	43.0	39.6	42.3	49.5	37.0	-	**54.2**	36.8	**57.3**
	$\text{mAP}^r_{0.5}$	15.6	2.7	25.7	22.4	17.0	21.6	12.5	-	26.0	**35.2**	**55.7**

tently outperforms the state-of-the-arts by a large margin. Especially, the Deepco3 obtains $\text{mAP}^r_{0.5}$ scores of 21.1, 12.3, 16.7 and 26.0 on COCO-VOC, COCO-NONVOC, VOC12 and SOC, while our GCAGC-ICS achieves the highest scores of 86.9, 76.8, 75.4 and 55.7 with a significant gain of 65.8%, 64.5%, 58.7% and 29.7%, respectively.

3) Ablative Studies: Here, ablative studies are conducted to confirm the positive affects of GCAGC-CSD in promoting performance of GCAGC-ICS. Tables III and IV list the statistic results of the proposed salient instance segmentation network (SIS) in Figure 1(e) and GCAGC-ICS. Specifically, GCAGC-ICS's scores of $\text{mAP}^r_{0.25}$ are 0.659, 0.585, and 0.671 in three sub-classes of *Ball*, *Cosm*, and *Traf*, which obviously outperform the scores 0.393, 0.442 and 0.513 of our proposed SIS in Table III. Moreover, all scores of GCAGC-ICS in terms of $\text{mAP}^r_{0.25}$ and $\text{mAP}^r_{0.5}$ are higher than the designed SIS, which sufficiently verify the positive influence of our GCAGC-CSD to enhance the performance of GCAGC-ICS.

E. Potential Applications

In many computer vision tasks, it is difficult to obtain strong supervision contexts like fine segmented ground-truth labels due to expensive human annotation costs, thus it is highly expected to explore the techniques in weakly supervised learning methods [95]–[99]. For the proposed GCAGC, it is also reasonable to establish the associated interactions of objects with common category and enhance the foreground importance in a single image by our AGCN and AGCM modules without leveraging fully supervised information. It will be a potential research topic in the future work.

V. Conclusion

This paper has presented an adaptive graph convolutional network framework with attention graph clustering for CSD and ICS tasks, mainly including two key designs: an AGCN and an AGCM. The AGCN has been developed to extract long-range dependency cues to characterize the intra- and inter-image correspondence. Meanwhile, to further refine the results of the AGCN, the AGCM has been designed to discriminate the co-objects from all the salient foreground objects in an unsupervised fashion. In general, a unified CSD framework with encoder-decoder structure has been implemented to jointly optimize the AGCN and the AGCM in an end-to-end manner. Moreover, we have designed a salient instance segmentation network, which is integrated into the CSD framework to realize the ICS task. Extensive evaluations on four CSD benchmark datasets (iCoseg, Cosal2015, COCO-SEG and CoSOD3k) and five ICS benchmark datasets (CoSOD3k, COCO-NONVOC, COCO-VOC, VOC12 and SOC) have demonstrated favorable performance of the proposed methods over the state-of-the-art methods in terms of most metrics.

References

[1] R. Cong, J. Lei, H. Fu, M.-M. Cheng, W. Lin, and Q. Huang, "Review of visual saliency detection with comprehensive information," *IEEE Transactions on circuits and Systems for Video Technology*, vol. 29, no. 10, pp. 2941–2959, 2018.

[2] K. R. Jerripothula, J. Cai, and J. Yuan, "Image co-segmentation via saliency co-fusion," *IEEE Transactions on Multimedia*, vol. 18, no. 9, pp. 1896–1909, 2016.

[3] C.-C. Tsai, W. Li, K.-J. Hsu, X. Qian, and Y.-Y. Lin, "Image co-saliency detection and co-segmentation via progressive joint optimization," *IEEE Transactions on Image Processing*, vol. 28, no. 1, pp. 56–71, 2018.

[4] H. Fu, D. Xu, S. Lin, and J. Liu, "Object-based rgbd image co-segmentation with mutex constraint," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2015, pp. 4428–4436.

[5] D.-P. Fan, W. Wang, M.-M. Cheng, and J. Shen, "Shifting more attention to video salient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 8554–8564.

[6] K. Fu, D.-P. Fan, G.-P. Ji, and Q. Zhao, "Jl-dcf: Joint learning and densely-cooperative fusion framework for rgb-d salient object detection,"

in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3052–3062.

[7] J.-X. Zhao, Y. Cao, D.-P. Fan, M.-M. Cheng, X.-Y. Li, and L. Zhang, "Contrast prior and fluid pyramid integration for rgbd salient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3927–3936.

[8] K. Tang, A. Joulin, L.-J. Li, and L. Fei-Fei, "Co-localization in real-world images," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2014, pp. 1464–1471.

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

[10] L. Yang, B. Geng, Y. Cai, A. Hanjalic, and X.-S. Hua, "Object retrieval using visual query context," *IEEE Transactions on multimedia*, vol. 13, no. 6, pp. 1295–1307, 2011.

[11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[12] H. Fu, X. Cao, and Z. Tu, "Cluster-based co-saliency detection," *IEEE Transactions on Image Processing*, vol. 22, no. 10, pp. 3766–3778, 2013.

[13] Z. Liu, W. Zou, L. Li, L. Shen, and O. Le Meur, "Co-saliency detection based on hierarchical segmentation," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 88–92, 2013.

[14] L. Ye, Z. Liu, J. Li, W.-L. Zhao, and L. Shen, "Co-saliency detection via co-salient object discovery and recovery," *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 2073–2077, 2015.

[15] C. Wang, Z.-J. Zha, D. Liu, and H. Xie, "Robust deep co-saliency detection with group semantic," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 8917–8924.

[16] D. Zhang, D. Meng, and J. Han, "Co-saliency detection via a self-paced multiple-instance learning framework," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 5, pp. 865–878, 2016.

[17] D. Zhang, J. Han, C. Li, and J. Wang, "Co-saliency detection via looking deep and wide," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2015, pp. 2994–3002.

[18] K. Zhang, T. Li, B. Liu, and Q. Liu, "Co-saliency detection via mask-guided fully convolutional networks with multi-scale label smoothing," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3095–3104.

[19] K.-J. Hsu, C.-C. Tsai, Y.-Y. Lin, X. Qian, and Y.-Y. Chuang, "Unsupervised cnn-based co-saliency detection with graphical optimization," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 485–501.

[20] L. Wei, S. Zhao, O. E. F. Bourahla, X. Li, and F. Wu, "Group-wise deep co-saliency detection," *arXiv preprint arXiv:1707.07381*, 2017.

[21] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2018, pp. 7794–7803.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[23] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2961–2969.

[24] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, "icoseg: Interactive co-segmentation with intelligent scribble guidance," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2010, pp. 3169–3176.

[25] D.-P. Fan, Z. Lin, G.-P. Ji, D. Zhang, H. Fu, and M.-M. Cheng, "Taking a deeper look at co-salient object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2919–2929.

[26] K.-J. Hsu, Y.-Y. Lin, and Y.-Y. Chuang, "Deepco3: Deep instance co-segmentation by co-peak search and co-saliency detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 8846–8855.

[27] K. Zhang, T. Li, S. Shen, B. Liu, J. Chen, and Q. Liu, "Adaptive graph convolutional network with attention graph clustering for co-saliency detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9050–9059.

[28] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[29] J. Lei, B. Wang, Y. Fang, W. Lin, P. Le Callet, N. Ling, and C. Hou, "A universal framework for salient object detection," *IEEE Transactions on Multimedia*, vol. 18, no. 9, pp. 1783–1795, 2016.

[30] J. Li, Z. Pan, Q. Liu, and Z. Wang, "Stacked u-shape network with channel-wise attention for salient object detection," *IEEE Transactions on Multimedia*, 2020.

[31] Q. Ren, S. Lu, J. Zhang, and R. Hu, "Salient object detection by fusing local and global contexts," *IEEE Transactions on Multimedia*, 2020.

[32] G. Li, Z. Liu, and H. Ling, "Icnet: Information conversion network for rgb-d based salient object detection," *IEEE Transactions on Image Processing*, vol. 29, pp. 4873–4884, 2020.

[33] C. Li, R. Cong, Y. Piao, Q. Xu, and C. C. Loy, "Rgb-d salient object detection with cross-modality modulation and selection," *arXiv preprint arXiv:2007.07051*, 2020.

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

[35] Y. Li, K. Fu, Z. Liu, and J. Yang, "Efficient saliency-model-guided visual co-saliency detection," *IEEE Signal Processing Letters*, vol. 22, no. 5, pp. 588–592, 2014.

[36] R. Cong, J. Lei, H. Fu, Q. Huang, X. Cao, and N. Ling, "Hscs: Hierarchical sparsity based co-saliency detection for rgbd images," *IEEE Transactions on Multimedia*, vol. 21, no. 7, pp. 1660–1671, 2018.

[37] C.-C. Tsai, K.-J. Hsu, Y.-Y. Lin, X. Qian, and Y.-Y. Chuang, "Deep co-saliency detection via stacked autoencoder-enabled fusion and self-trained cnns," *IEEE Transactions on Multimedia*, vol. 22, no. 4, pp. 1016–1031, 2019.

[38] X. Cao, Z. Tao, B. Zhang, H. Fu, and W. Feng, "Self-adaptively weighted co-saliency detection via rank constraint," *IEEE Transactions on Image Processing*, vol. 23, no. 9, pp. 4175–4186, 2014.

[39] H. Song, Z. Liu, Y. Xie, L. Wu, and M. Huang, "Rgbd co-saliency detection via bagging-based clustering," *IEEE Signal Processing Letters*, vol. 23, no. 12, pp. 1722–1726, 2016.

[40] J. Ren, Z. Liu, X. Zhou, C. Bai, and G. Sun, "Co-saliency detection via integration of multi-layer convolutional features and inter-image propagation," *Neurocomputing*, vol. 371, pp. 137–146, 2020.

[41] J. Ren, Z. Liu, G. Li, X. Zhou, C. Bai, and G. Sun, "Co-saliency detection using collaborative feature extraction and high-to-low feature integration," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2020, pp. 1–6.

[42] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, vol. 2, 2005, pp. 729–734.

[43] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2008.

[44] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, and M. Sun, "Graph neural networks: A review of methods and applications," *arXiv preprint arXiv:1812.08434*, 2018.

[45] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," *arXiv preprint arXiv:1312.6203*, 2013.

[46] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Advances in Neural Information Processing Systems*, 2016, pp. 3844–3852.

[47] J. Atwood and D. Towsley, "Diffusion-convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 1993–2001.

[48] M. Niepert, M. Ahmed, and K. Kutzkov, "Learning convolutional neural networks for graphs," in *International Conference on Machine Learning*, 2016, pp. 2014–2023.

[49] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh, "Graph r-cnn for scene graph generation," in *European conference on computer vision*, 2018, pp. 670–685.

[50] L. Landrieu and M. Simonovsky, "Large-scale point cloud semantic segmentation with superpoint graphs," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2018, pp. 4558–4567.

[51] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 5, p. 146, 2019.

[52] W. Wang, X. Lu, J. Shen, D. J. Crandall, and L. Shao, "Zero-shot video object segmentation via attentive graph neural networks," in *International Conference on Computer Vision*, 2019, pp. 9236–9245.

[53] X. Qi, R. Liao, J. Jia, S. Fidler, and R. Urtasun, "3d graph neural networks for rgbd semantic segmentation," in *International Conference on Computer Vision*, 2017, pp. 5199–5208.

[54] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Thirty-second AAAI conference on artificial intelligence*, 2018.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2021.3054526, IEEE Transactions on Multimedia

[55] X. Chen, L.-J. Li, L. Fei-Fei, and A. Gupta, "Iterative visual reasoning beyond convolutions," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2018, pp. 7239–7248.

[56] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *arXiv preprint arXiv:1901.00596*, 2019.

[57] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[58] Y. Sun, J. Han, J. Gao, and Y. Yu, "itopicmodel: Information network-integrated topic modeling," in *2009 Ninth IEEE International Conference on Data Mining*, 2009, pp. 493–502.

[59] X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, and S. Yang, "Community preserving network embedding," in *Thirty-first AAAI conference on artificial intelligence*, 2017.

[60] C. Wang, S. Pan, R. Hu, G. Long, J. Jiang, and C. Zhang, "Attributed graph clustering: A deep attentional embedding approach," *IJCAI*, 2019.

[61] S. Pan, R. Hu, S.-f. Fung, G. Long, J. Jiang, and C. Zhang, "Learning graph embedding with adversarial training methods," *arXiv preprint arXiv:1901.01250*, 2019.

[62] D. A. Bader, H. Meyerhenke, P. Sanders, and D. Wagner, *Graph partitioning and graph clustering*. American Mathematical Soc., 2013, vol. 588.

[63] G. Li, Y. Xie, L. Lin, and Y. Yu, "Instance-level salient object segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2017, pp. 2386–2395.

[64] R. Fan, M.-M. Cheng, Q. Hou, T.-J. Mu, J. Wang, and S.-M. Hu, "S4net: Single stage salient-instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6103–6112.

[65] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3150–3158.

[66] Y. Zhou, Y. Zhu, Q. Ye, Q. Qiu, and J. Jiao, "Weakly supervised instance segmentation using class peak response," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2018, pp. 3791–3800.

[67] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[68] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[69] Q. Li, Z. Han, and X.-M. Wu, "Deeper insights into graph convolutional networks for semi-supervised learning," in *Thirty-second AAAI conference on artificial intelligence*, 2018.

[70] Q. Li, X.-M. Wu, H. Liu, X. Zhang, and Z. Guan, "Label efficient semi-supervised learning via graph filtering," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9582–9591.

[71] X. Zheng, Z.-J. Zha, and L. Zhuang, "A feature-adaptive semi-supervised framework for co-saliency detection," in *MM*, 2018, pp. 959–966.

[72] B. Jiang, X. Jiang, A. Zhou, J. Tang, and B. Luo, "A unified multiple graph learning and convolutional network model for co-saliency estimation," in *MM*, 2019, pp. 1375–1382.

[73] M. Henaff, J. Bruna, and Y. LeCun, "Deep convolutional networks on graph-structured data," *arXiv preprint arXiv:1506.05163*, 2015.

[74] R. Li, S. Wang, F. Zhu, and J. Huang, "Adaptive graph convolutional neural networks," *arXiv preprint arXiv:1801.03226*, 2018.

[75] X. Wang and A. Gupta, "Videos as space-time region graphs," in *European conference on computer vision*, 2018, pp. 399–417.

[76] I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted graph cuts without eigenvectors a multilevel approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 11, pp. 1944–1957, 2007.

[77] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[78] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2009, pp. 248–255.

[79] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, 2014, pp. 740–755.

[80] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[81] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," *IEEE Transactions on Pattern analysis and machine intelligence*, vol. 33, no. 2, pp. 353–367, 2010.

[82] D. Zhang, H. Fu, J. Han, A. Borji, and X. Li, "A review of co-saliency detection algorithms: fundamentals, applications, and challenges," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 9, no. 4, p. 38, 2018.

[83] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *International Conference on Computer Vision*, 2017, pp. 4548–4557.

[84] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, and A. Borji, "Enhanced-alignment measure for binary foreground map evaluation," in *IJCAI*, 2018.

[85] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *European conference on computer vision*. Springer, 2014, pp. 297–312.

[86] J. Han, G. Cheng, Z. Li, and D. Zhang, "A unified metric learning-based framework for co-saliency detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 10, pp. 2473–2483, 2017.

[87] D. Zhang, J. Han, J. Han, and L. Shao, "Cosaliency detection based on intrasaliency prior transfer and deep intersaliency mining," *TNNLS*, vol. 27, no. 6, pp. 1163–1176, 2015.

[88] Z. Zhang, W. Jin, J. Xu, and M.-M. Cheng, "Gradient-induced co-saliency detection," *European conference on computer vision*, 2020.

[89] M. Cho, S. Kwak, C. Schmid, and J. Ponce, "Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2015, pp. 1201–1210.

[90] X.-S. Wei, C.-L. Zhang, Y. Li, C.-W. Xie, J. Wu, C. Shen, and Z.-H. Zhou, "Deep descriptor transforming for image co-localization," *IJCAI*, 2017.

[91] X.-S. Wei, C.-L. Zhang, J. Wu, C. Shen, and Z.-H. Zhou, "Unsupervised object discovery and co-localization by deep descriptor transformation," *Pattern Recognition*, vol. 88, pp. 113–126, 2019.

[92] E. Collins, R. Achanta, and S. Susstrunk, "Deep feature factorization for concept discovery," in *ECCV*, 2018, pp. 336–352.

[93] Z. Luo, A. Mishra, A. Achkar, J. Eichel, S. Li, and P.-M. Jodoin, "Non-local deep features for salient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2017, pp. 6609–6617.

[94] X. Li, F. Yang, H. Cheng, W. Liu, and D. Shen, "Contour knowledge transfer for salient object detection," in *European conference on computer vision*, 2018, pp. 355–370.

[95] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, "Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 6, pp. 3325–3337, 2014.

[96] D. Zhang, J. Han, L. Yang, and D. Xu, "Spftn: a joint learning framework for localizing and segmenting objects in weakly labeled videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[97] D. Zhang, J. Han, L. Zhao, and D. Meng, "Leveraging prior-knowledge for weakly supervised object detection under a collaborative self-paced curriculum learning framework," *International Journal of Computer Vision*, vol. 127, no. 4, pp. 363–380, 2019.

[98] D. Zhang, J. Han, G. Guo, and L. Zhao, "Learning object detectors with semi-annotated weak labels," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 12, pp. 3622–3635, 2018.

[99] X. Li, M. Kan, S. Shan, and X. Chen, "Weakly supervised object detection with segmentation collaboration," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9735–9744.

Tengpeng Li received his B.S. and M.S. degrees from the School of Automation in Nanjing University of Information Science & Technology in Jun. 2017 and Jun. 2020, respectively. His research interests include machine learning and computer vision, especially on saliency detection, co-salient object detection and weakly supervised learning.

Kaihua Zhang is a Professor in the School of Automation, Nanjing University of Information Science & Technology, Nanjing, China. He received the B.S. degree in Technology and Science of Electronic Information from Ocean University of China (OUC) in 2006, the M.S. degree in Signal and Information Processing from the University of Science and Technology of China (USTC) in 2009 and Ph.D degree from the Department of Computing in the Hong Kong Polytechnic University in 2013. From Aug. 2009 to Aug. 2010, he worked as a Research Assistant in the Department of Computing, The Hong Kong Polytechnic University. His research interests include image segmentation, level sets, and visual tracking.

Shiwen Shen is a senior research scientist at JD Finance America Corporation. He obtained his Ph.D degree in Medical Imaging Informatics Group of University of Carlifornia, Los Angeles (UCLA). Before that, he obtained his Master's degree in Electrical Engineering from Shanghai Jiao Tong University. His research interests include image and video understanding, medical image analysis, longitidual data analysis, and Bayesian Modeling.

Bo Liu is a research scientist at JD Finance America Corporation. His current research focuses on machine learning, computer vision and data analytics. He received PhD degree from the Computer Science Department, Rutgers, The State University of New Jersey in 2018. Before that he worked as a research staff at The Hong Kong Polytechnic University. His other previous employments include Siemens Healthineers, GE Global Research and Microsoft Research Asia.

Qingshan Liu is a Professor with the School of Automation, Nanjing University of Information Science and Technology, Nanjing, China. He received the Ph.D. degree from the National Laboratory of Pattern Recognition, Chinese Academic of Science, Beijing, China, in 2003, and the M.S. degree from the Department of Auto Control, Southeast University, Nanjing, in 2000. He was an Assistant Research Professor with the Department of Computer Science, Computational Biomedicine Imaging and Modeling Center, Rutgers, The State University of New Jersey, New Brunswick, NJ, USA, from 2010 to 2011. Before he joined Rutgers University, he was an Associate Professor with the National Laboratory of Pattern Recognition, Chinese Academic of Science, and an Associate Researcher with the Multimedia Laboratory, Chinese University of Hong Kong, Hong Kong, from 2004 and 2005. He was a recipient of the President Scholarship of the Chinese Academy of Sciences in 2003. His current research interests are image and vision analysis, including face image analysis, graph and hypergraph-based image and video understanding, medical image analysis, and event-based video analysis.

Zhu Li now an Associate Professor with the Dept of Computer Science & Electrical Engnieering (CSEE), University of Missouri,Kansas City, and director of the NSF I/UCRC Center for Big Learning (CBL) at UMKC. He received his PhD in Electrical & Computer Engineering from Northwestern University, Evanston in 2004. He was AFOSR SFFP summer visiting faculty at the US Air Force Academy (USAFA), 2016 , 2017 , 2018 and 2020, with the UAV Research Center. He was Sr. Staff Researcher/Sr. Manager with Samsung Research America's Multimedia Standards Research Lab in Richardson, TX, 2012-2015, Sr. Staff Researcher/Media Analytics Lead with FutureWei (Huawei) Technology's Media Lab in Bridgewater, NJ, 2010 2012, an Assistant Professor with the Dept of Computing, The Hong Kong Polytechnic University from 2008 to 2010, and a Principal Staff Research Engineer with the Multimedia Research Lab (MRL), Motorola Labs, from 2000 to 2008. His research interests include point cloud and light field compression, graph signal processing and deep learning in the next gen visual compression, image processing and understanding. He has 47 issued or pending patents, 100+ publications in book chapters, journals, and conferences in these areas. He is an IEEE senior member, associate Editor-in-Chief for IEEE Trans on Circuits & System for Video Tech, associated editor for IEEE Trans on Image Processing(2020), IEEE Trans.on Multimedia (2015-18), IEEE Trans on Circuits & System for Video Technology(2016-19). He serves on the steering committee member of IEEE ICME (2015-18), he is an elected member of the IEEE Multimedia Signal Processing (MMSP), IEEE Image, Video, and Multidimensional Signal Processing (IVMSP), and IEEE Visual Signal Processing & Communication (VSPC) Tech Committees. He is program co-chair for IEEE Intl Conf on Multimedia & Expo (ICME) 2019, and co-chaired the IEEE Visual Communication & Image Processing (VCIP) 2017. He received the Best Paper Award at IEEE Int'l Conf on Multimedia & Expo (ICME), Toronto, 2006, the Best Paper Award (DoCoMo Labs Innovative Paper) at IEEE Int'l Conf on Image Processing (ICIP), San Antonio, 2007.