Summarize and Search: Learning Consensus-aware Dynamic Convolution for Co-Saliency Detection

Ni Zhang[1] Junwei Han[1] Nian Liu[2*] Ling Shao[2]
[1]Northwestern Polytechnical University [2]Inception Institute of Artificial Intelligence
{nnizhang.1995, junweihan2010, liunian228}@gmail.com, ling.shao@ieee.org

Abstract

Humans perform co-saliency detection by first summarizing the consensus knowledge in the whole group and then searching corresponding objects in each image. Previous methods usually lack robustness, scalability, or stability for the first process and simply fuse consensus features with image features for the second process. In this paper, we propose a novel consensus-aware dynamic convolution model to explicitly and effectively perform the "summarize and search" process. To summarize consensus image features, we first summarize robust features for every single image using an effective pooling method and then aggregate cross-image consensus cues via the self-attention mechanism. By doing this, our model meets the scalability and stability requirements. Next, we generate dynamic kernels from consensus features to encode the summarized consensus knowledge. Two kinds of kernels are generated in a supplementary way to summarize fine-grained image-specific consensus object cues and the coarse group-wise common knowledge, respectively. Then, we can effectively perform object searching by employing dynamic convolution at multiple scales. Besides, a novel and effective data synthesis method is also proposed to train our network. Experimental results on four benchmark datasets verify the effectiveness of our proposed method. Our code and saliency maps are available at `https://github.com/nnizhang/CADC`.

1. Introduction

Co-salient object detection (Co-SOD) mimics the human visual system to distinguish common and salient objects when viewing a group of relevant images. Although various Co-SOD methods have been proposed, let us review this problem from the humans perspective. Given a group of images, humans can not segment the co-salient object in each image directly. Instead, they need to first observe all images and summarize the consensus knowledge about what kind of objects this group is focusing on. Then, they look back at each image and search the corresponding objects. We call this process "*summarize and search*", which is illustrated in Figure 1. A similar explanation can also be found in [48]. Therefore, we can model Co-SOD in such an intuitive way to summarize the consensus knowledge first and then search consensus objects in each image.

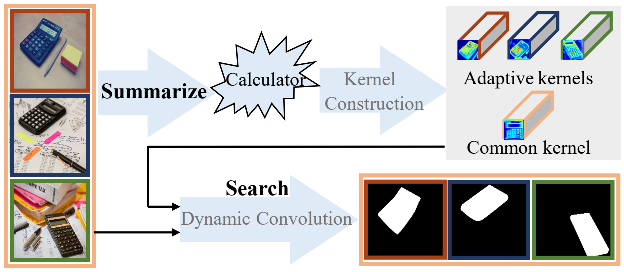

Figure 1. Main idea of our proposed method.

Previous models can also be explained from such a point. For consensus knowledge summarization, early traditional methods employed graph models [18] or clustering methods [7, 39] to learn the common patterns. However, their models lack end-to-end learning, thus limiting the model performance. Some recent deep models [36, 35, 27] chose to concatenate and convolve all image features for summarizing the consensus knowledge. However, convolution can only aggregate the information at the same location among different images, while co-salient objects often show variations in scales and locations in different images. Hence, these models may easily fail in consensus summarization. Using non-local dependencies [34] to summarize the consensus cues is another choice [8]. However, this method lacks scalability since it is computationally prohibitive for processing a large number of images. Some other work [17] adopted recurrent networks to summarize the consensus cues step by step. However, recurrent models define an input order for image sequences, thus lacking model stability since different input orders will lead to different results.

For consensus object searching, many works [36, 35, 17, 32, 40, 27, 47] directly fused the consensus feature

[*]Corresponding author.

with image-specific features via summation or concatenation operations. [46] and [5] fused co-attention maps with the image-specific information via element-wise multiplication. Such simple methods conduct object searching by linear information fusion, which can not fully exploit the guidance of the summarized consensus knowledge. Besides, [48] computed channel-wise weight for each single image feature based on its similarity with the consensus representation, which can be seen as an attribute-wise object searching method. We argue that direct spatial searching might be more accurate and easy to learn.

In this paper, we propose a novel consensus-aware dynamic convolution (CADC) model directly from the "summarize and search" point of view. The image features of the whole group are first summarized and then the consensus knowledge is encoded as dynamic kernels, which capture the appearance traits of common objects. Next, the searching step is performed by using the kernels to convolve the image features to obtain final results, as shown in Figure 1.

However, adopting dynamic convolution for Co-SOD requires delicate model design. We propose to summarize the consensus knowledge via first summarizing the feature of every single image and then integrating cross-image consensus features. For the first step, we propose to use a multi-scale max-pooling module to achieve position and scale robust features. For the second step, we leverage the self-attention mechanism [31]. In this way, our model can meet the needs for scalability and stability. For consensus-aware dynamic kernel generation, we propose to simultaneously construct image adaptive kernels and a common kernel. The former is generated for each image separately to capture fine-grained image-specific cues while the latter is generated for the whole group to summarize coarse group-wise common knowledge. Theoretically, the latter can serve as a supplement and regularization for the former to avoid them focusing too much on the image-specific information. We also generate efficient large dynamic kernels to further consider spatial structures and enlarge the searching range.

Besides, considering the lack of training data in the Co-SOD field, we propose a novel and effective data synthesis method by fusing common objects with unrelated salient objects in two different ways to mimic the real-world scenarios. It can largely improve the Co-SOD performance and shows superiority when compared with previous methods.

Our main contributions can be summarized as follows.

- From the "summarize and search" perspective, we propose a novel CADC model for Co-SOD. Dynamic kernels are generated to summarize the consensus knowledge and object searching is performed using dynamic convolution.
- We propose to combine multi-scale max-pooling and self-attention models to obtain consensus features with both model scalability and stability.
- We construct two types of dynamic kernels in a supplementary way to capture image-specific cues and the group-wise common knowledge, respectively.
- We develop a novel and more effective data synthesis method to mimic the challenging scenarios in the real world for Co-SOD models training.
- Our CADC network achieves new state-of-the-art Co-SOD results.

2. Related Work

2.1. Co-Saliency Detection

Early Co-SOD methods [7, 18] often devote to design hand-crafted features based on different low-level image features. Recent works have introduced deep learning techniques into Co-SOD and gained promising performance. One bunch of works [44, 46] combines deep learning features with other traditional methods. However, such separate learning schemes do not make full use of the advantage of CNNs in a data-driven manner.

In contrast, another bunch of works adopts end-to-end deep models to learn the common patterns of relevant images. [36, 35, 27] concatenated and convolved all image features to generate the consensus feature, which is sensitive to the variations of object locations and scales. In contrast, we propose a multi-scale max-pooling module to extract position and scale robust features. Besides, some works [36, 35, 17, 32, 40, 27, 47] employed summation or concatenation operations to fuse the consensus features with single image features in a linear space. As a result, they can not explore more effective guidance from the consensus knowledge, hence performing unsatisfactorily for object searching. In contrast, we learn two types of consensus-aware dynamic kernels to perform diverse and supplementary consensus summarization and perform dynamic convolution for effective object searching.

Some other existing models explore long-range dependencies to detect co-salient objects, such as [8, 17]. However, [8] only explored the interaction between a pair of images by the non-local network [34], which is fragile to obtain common features because similar extraneous objects may also appear in both images and cause distraction. Besides, this method also lacks scalability since it is computationally prohibitive to process a large number of images. [17] utilized recurrent networks to explore the interactions from all available images step by step. However, recurrent models have a sequential order issue and may cause model instability. In contrast, our adopted multi-scale max-pooling module can first decrease the feature dimensionality for each image, which further enables us to summarize the consensus knowledge from all images through the self-attention mechanism. Thus, we can effectively capture the global consensus with both model scalability and stability.

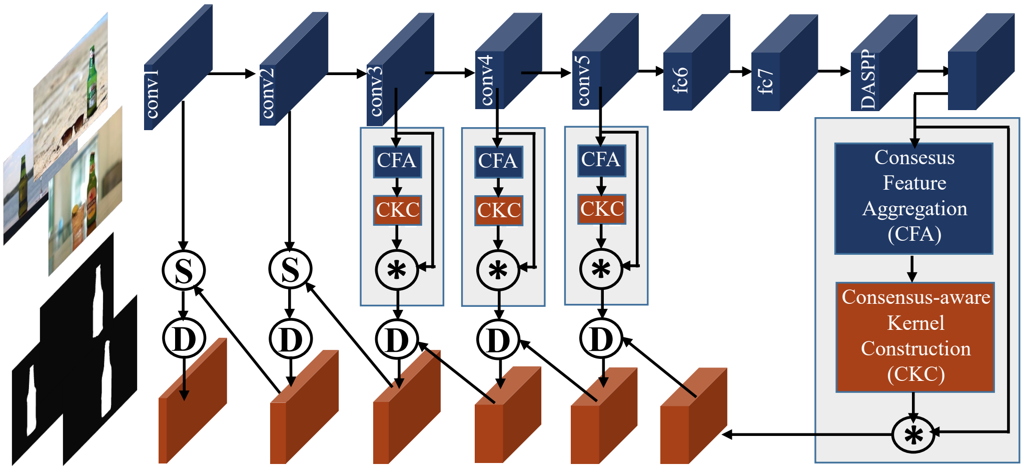
conv1
conv2
conv3
conv4
conv5
fc6
fc7
DASPP
CFA
CKC
S
D
Consesus Feature Aggregation (CFA)
Consensus-aware Kernel Construction (CKC)

Figure 2. Framework of our CADC network. ⊛ and Ⓓ mean dynamic convolution and our decoder module, respectively. Ⓢ denotes the spatial attention.

2.2. Dynamic Convolution

As a specified method of meta-learning, dynamic convolution uses predicted kernels to perform the convolution operation, which is different from traditional convolutions with fixed filters once trained. Xu *et al.* [15] proposed a dynamic filter network to learn custom parameters for different input samples. This idea is widely adopted to address the few-shot learning problem [1, 9], where a learner is first trained on a large set of available training data of base categories and then utilized to generate dynamic weights for classifying novel categories. Subsequently, some works [26, 30] introduced dynamic convolution into the instance segmentation task. However, most of them only learned one dynamic kernel with the 1×1 size and did not consider learning large spatial kernels due to the involved large computational costs. Pang *et al.* [23] introduced large 3×3 dynamic kernels with different dilation rates for RGB-D SOD. However, they generated a different kernel for every pixel in every image, which has significantly large computational costs. Different from previous methods, we design both group-specific and image-specific dynamic kernels to learn diverse and supplementary meta knowledge for Co-SOD. Our dynamic convolution is also computationally efficient by using the depthwise separable mechanism [12].

3. Proposed Method

Figure 2 shows our overall pipeline for Co-SOD. We propose the CADC model for consensus summarization and object searching, where the former is performed by consensus feature aggregation and consensus-aware kernel construction. We embed this model into a U-shaped [28] model and conduct hierarchical object searching in multiple feature levels. At the same time, we propose a novel and effective data synthesis method to train the proposed network.

3.1. Consensus Feature Aggregation

Given a group of N relevant images $\{I^n\}_{n=1}^N$, we first employ our encoder to extract their encoding feature maps $\boldsymbol{X} \in \mathbb{R}^{N \times H \times W \times C}$, where H, W, and C represent its height, width, and channel number, respectively. We follow [22] to slightly modify the original VGG-16 [29] backbone and insert the modified DASPP module [38] after it as our encoder. Then, as shown in Figure 3, we employ the adaptive max-pooling layer on $\boldsymbol{X}$ with three target scales and obtain the output features with spatial sizes of 1×1, 3×3 and 6×6, respectively. Then, these output features are flattened and concatenated to generate a feature $\boldsymbol{F} \in \mathbb{R}^{N \times 46 \times C}$. For each image, the obtained feature summarizes the dominated object features at multiple scales, hence is robust to both position and scale variations of co-salient objects.

The multi-scale max-pooling module dramatically decreases the feature number of each image from $H \times W$ to 46, hence providing us the feasibility to summarize the global consensus from all images by self-attention. Following [31], we first apply linear transformations to project $\boldsymbol{F}$ into the query, key, and value spaces with $\frac{C}{2}$ channels. After that, an affinity matrix $\boldsymbol{A} \in \mathbb{R}^{46N \times 46N}$ is calculated by matrix multiplication between the query and the key matrices, and it indicates the pairwise similarities among the $46N$ features of all images. Since a feature is usually more similar to other features in the same image than those in other images, we reset the self-similarity elements in $\boldsymbol{A}$ computed within each same image into a very small value to avoid intra-image similarities dominating the affinity matrix.

After that, we obtain an attention matrix via adopting normalization along the second dimension and then perform matrix multiplication with the value to generate an aggregated feature $\boldsymbol{Y} \in \mathbb{R}^{46N \times \frac{C}{2}}$. Next, $\boldsymbol{Y}$ is re-projected to C channels by a linear transformation and then reshaped to the shape $N \times 46 \times C$. Finally, it is added onto the original

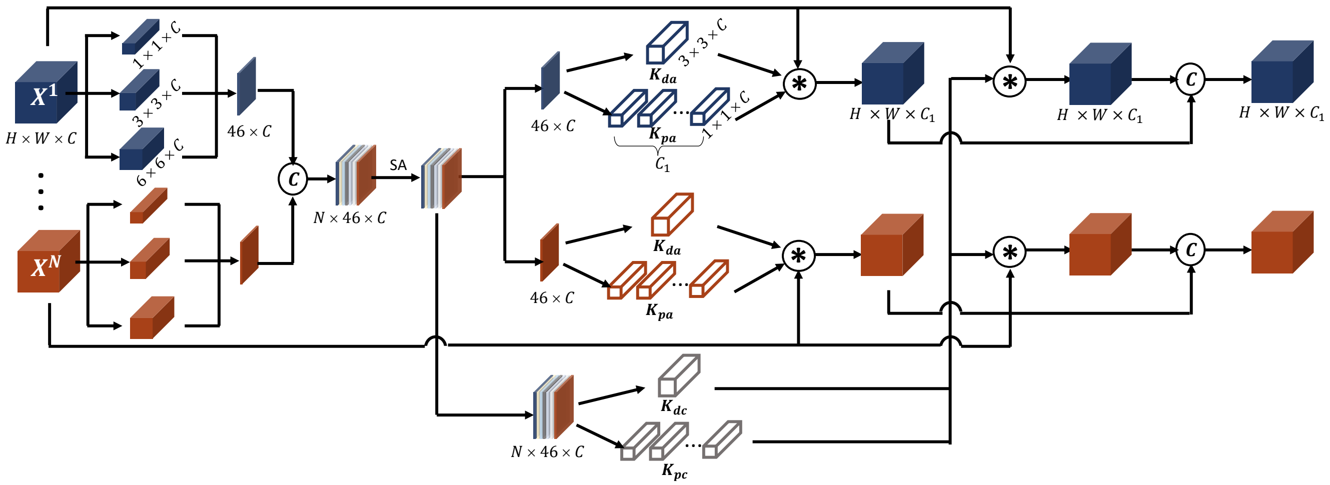

Figure 3. Pipeline of our proposed CADC for consensus summarization and object searching. We generate two types of kernels, *i.e*., adaptive kernel, and common kernel, for each image and the whole group, respectively. 'SA' means the self-attention module. ⊛ and © mean the depthwise separable convolution and concatenation, respectively.

feature map $\boldsymbol{F}$ for providing a residual signal to generate the consensus feature $\boldsymbol{Z} \in \mathbb{R}^{N\times 46\times C}$.

3.2. Consensus-aware Kernel Construction

Based on the consensus feature $\boldsymbol{Z}$, we generate two kinds of kernels for each image group to encode the summarized consensus knowledge. Since the co-occurring salient objects may have various appearances and scales in different images, we first construct adaptive kernels for each image to encode the fine-grained image-specific consensus object information. A common kernel is also generated for the whole group to capture the coarse group-wise common object knowledge. The latter can be regarded as a supplement and regularization for the former to avoid them paying too much attention to the image-specific information and ignoring the common information. To this end, generating these two kinds of kernels disentangles the learning of image-specific consensus object information and group-wise common object knowledge, thus better conforming to the nature of Co-SOD and facilitating potential relation exploration between them. Besides, doing so mimics the multi-branch architecture widely used in CNNs, which increases the transformation complexity and model capability.

(1) Vanilla dynamic kernels with 1×1 size

We first follow most traditional dynamic convolution methods [26, 30] to generate dynamic kernels with 1×1 size, which is straightforward and easy to implement.

Adaptive kernels with 1×1 size. We utilize $\boldsymbol{Z}$ to generate adaptive kernels for different images. First, we flatten $\boldsymbol{Z}$ to $\mathbb{R}^{N\times 46C}$ and learn a feature attention $\boldsymbol{\alpha} \in \mathbb{R}^{N\times 46}$ via:

$$\boldsymbol{\alpha} = FC(ReLU(BN(FC(\boldsymbol{Z})))), \tag{1}$$

where $\boldsymbol{\alpha}$ is further normalized by the softmax operation along the second dimension to select which one is the most discriminative among all the 46 features of each image. The intermediate FC layer has 1024 nodes.

Next, $\boldsymbol{Z}$ is weighted summed by $\boldsymbol{\alpha}$ along the second dimension to generate the attended feature $\boldsymbol{F_a} \in \mathbb{R}^{N\times C}$.

Finally, the 1×1 adaptive kernels are learned from $\boldsymbol{F_a}$ via:

$$\boldsymbol{K_a} = FC(PReLU(BN(FC(\boldsymbol{F_a})))), \tag{2}$$

where $\boldsymbol{K_a} \in \mathbb{R}^{N\times C_1 C}$ and is further reshaped to the shape $N \times C_1 \times C \times 1 \times 1$. Here C_1 is the desired output channel number of the dynamic convolution operation and the intermediate FC layer has C nodes. We adopt the Parametric ReLU (PReLU) [11] activation function for generating kernels since they usually have both positive and negative activation.

Common Kernel with 1×1 size. We aim to employ an attention weight $\boldsymbol{W} \in \mathbb{R}^{N\times 46}$ to aggregate all image features in $\boldsymbol{Z}$ along the first two dimensions and generate a group-wise common feature $\boldsymbol{F_c} \in \mathbb{R}^{C}$. As discussed in [2], the computed self-attention of different queries all tend to highlight the same set of most discriminative key elements. Thus, we can find which features are the most discriminative from the self-attention matrix $softmax(\boldsymbol{A})$. Specifically, we can get the weight $\boldsymbol{W}$ by averaging this matrix along the first dimension. Then, $\boldsymbol{F_c}$ can be obtained by using $\boldsymbol{W}$ to weighted sum $\boldsymbol{Z}$ along the first two dimensions.

Finally, the common kernel can be learned via:

$$\boldsymbol{K_c} = FC(PReLU(FC(\boldsymbol{F_c}))), \tag{3}$$

where $\boldsymbol{K_c} \in \mathbb{R}^{C_1 C}$ and is further reshaped to the shape $C_1 \times C \times 1 \times 1$ as the group-wise kernel. The intermediate FC layer also has C nodes.

(2) Efficient large dynamic kernels

The vanilla dynamic kernels with 1×1 size can only encode channel-wise consensus knowledge and ignore spatial cues. Besides, they can only enable object searching within the 1×1 range, resulting in limited searching capability. To introduce spatial cues for the consensus knowledge and also enlarge the searching range, we propose to generate large spatial dynamic kernels. However, it will incur large computation costs and large amounts of the FC parameters if directly use the same method as the vanilla dynamic kernels. For instance, if we want to generate dynamic kernels with a spatial size of 3×3, $\boldsymbol{K_a}$ and $\boldsymbol{K_c}$ will be 9 times larger, so do the parameters of the FC layers used to generate them. This is also the reason that most dynamic convolution methods do not learn large spatial kernels. We overcome this issue by using the depthwise separable convolution [12] operation. We construct both adaptive kernels and the common kernel with the 3×3 size in this form as follows.

Adaptive Kernels with 3×3 size. We decompose 3×3 adaptive kernels $\boldsymbol{K_{la}}$ into depthwise adaptive kernels $\boldsymbol{K_{da}} \in \mathbb{R}^{N\times C\times3\times3}$ and pointwise adaptive kernels $\boldsymbol{K_{pa}} \in \mathbb{R}^{N\times C_1\times C\times1\times1}$. The latter can be constructed in the same way as $\boldsymbol{K_a}$.

To construct $\boldsymbol{K_{da}}$, We transform each of the 46-d features in $\boldsymbol{Z}$ to generate the 3×3 kernel for each channel and each image. We first permute $\boldsymbol{Z}$ to the shape $\mathbb{R}^{NC\times46}$ and adopt FC layers as follows:

$$\boldsymbol{K_{da}} = FC(PReLU(BN(FC(\boldsymbol{Z})))), \tag{4}$$

where the intermediate FC layer has 46 nodes, $\boldsymbol{K_{da}} \in \mathbb{R}^{NC\times9}$ and it is further reshaped to the shape $N \times C \times 3 \times 3$.

Common Kernel with 3×3 size. We also decompose the 3×3 common kernel into the depthwise common kernel $\boldsymbol{K_{dc}} \in \mathbb{R}^{C\times3\times3}$ and the pointwise common kernel $\boldsymbol{K_{pc}} \in \mathbb{R}^{C_1\times C\times1\times1}$. Note that the construction of $\boldsymbol{K_{pc}}$ is also the same as $\boldsymbol{K_c}$.

To construct $\boldsymbol{K_{dc}}$, we need to aggregate the information across N images in the consensus feature $\boldsymbol{Z}$. To this end, we leverage an attention $\boldsymbol{\alpha_3} \in \mathbb{R}^N$ to aggregate the image features with its N attention weights.

To learn $\boldsymbol{\alpha_3}$, we first flatten $\boldsymbol{Z}$ to the shape of $\mathbb{R}^{N\times46C}$ and then generate two attentions $\boldsymbol{\alpha_1} \in \mathbb{R}^{N\times C}$ and $\boldsymbol{\alpha_2} \in \mathbb{R}^{N\times46}$ via:

$$\begin{aligned} \boldsymbol{\alpha_1} &= FC(ReLU(BN(FC(\boldsymbol{Z})))), \\ \boldsymbol{\alpha_2} &= FC(ReLU(BN(FC(\boldsymbol{Z})))), \end{aligned} \tag{5}$$

where the intermediate FC layers for both $\boldsymbol{\alpha_1}$ and $\boldsymbol{\alpha_2}$ have 1024 nodes. Then, $\boldsymbol{\alpha_1}$ and $\boldsymbol{\alpha_2}$ are further normalized by softmax along the second dimension. Next, we can obtain $\boldsymbol{\alpha_3}$ by successively using $\boldsymbol{\alpha_1}$ and $\boldsymbol{\alpha_2}$ to aggregate the information in $\boldsymbol{Z}$ along the "C" and the "46" dimensions.

Finally, we apply softmax on $\boldsymbol{\alpha_3}$ and employ it to weighted sum $\boldsymbol{Z}$ for eliminating the first dimension. As a result, we can obtain a feature $\boldsymbol{F_{dc}} \in \mathbb{R}^{C\times46}$, which is utilized to generate $\boldsymbol{K_{dc}}$ via:

$$\boldsymbol{K_{dc}} = FC(PReLU(BN(FC(\boldsymbol{F_{dc}})))), \tag{6}$$

where the intermediate FC layer has 46 nodes, $\boldsymbol{K_{dc}} \in \mathbb{R}^{C\times9}$ and is further reshaped to the shape $C \times 3 \times 3$.

3.3. Object Searching via Dynamic Convolution

After obtaining consensus-aware dynamic kernels, we adopt dynamic convolution on the original feature maps $\{\boldsymbol{X^n}\}_{n=1}^N$ to perform explicit object searching. For vanilla 1×1 kernels, we directly use them to convolve each $\boldsymbol{X^n}$. For efficient large kernels, following depthwise separable convolution [12], we first use the depthwise kernels to perform 3×3 group convolution for each channel separately, and then adopt the pointwise kernels to perform regular 1×1 convolution.

For each image, we use both its adaptive kernel and the common kernel to perform dynamic convolution simultaneously and then fuse the two response maps to C_1 channels via concatenation and convolution, as shown in Figure 3.

We use the proposed CADC to connect the encoder-decoder pairs in our U-shaped network at multiple levels, hence performing hierarchical object searching at different scales and can effectively improve the searching accuracy. Specifically, we perform hierarchical object searching in the first four decoder modules. In each decoder module, we first perform CADC on the encoder feature map. Then, we concatenate the searching response map with the previous decoder feature map and use two 3×3 Conv layers to fuse them. BN [14] layers and ReLU are also used right after the Conv layers. For the last two decoder modules, we do not use CADC anymore. Instead, we simply use the previous decoder feature to generate a spatial attention map to filter the current encoder feature, as shown in Figure 2.

3.4. Computational Costs Analysis

In this section, we discuss the computational costs of our consensus feature aggregation and consensus-aware kernel construction. In the former, our multi-scale max-pooling module dramatically decreases the feature number of each image from $H \times W$ to 46, hence making it possible to aggregate a group of images while the original self-attention incurs large computational costs. For example, given N images, the computational complexity of using self-attention on the original feature maps and our pooled ones are $O((NWH)^2)$ and $O((46N)^2)$, respectively, where $46 \ll WH$. For the consensus-aware kernel construction, our

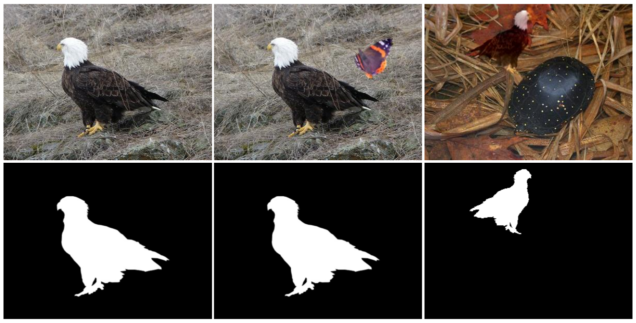

Figure 4. Examples of our proposed data synthesis method. The first and second columns show the original image and the normally synthesized image, and their corresponding ground truth. The last column shows the reversely synthesized image.

model enlarges the searching range without dramatically increasing computational costs by introducing the depthwise-separable convolution. It reduces the kernel size to construct from $C_1 \times C \times 3 \times 3$ to $C \times 3 \times 3 + C_1 \times C$.

4. New Data Synthesis Strategy

Many previous models [44, 10, 19, 17] combined various datasets to train their Co-SOD models. We follow [16] to use a subset of the COCO dataset [20] with 9213 images of 65 groups to train our model. However, this dataset highlights all objects that belong to the same category as ground truth, without discriminating salient and non-salient ones. To this end, [16] had to use an off-the-shelf SOD model [49] trained on DUTS [33] as a pre-computed saliency prior. Hence, we also leverage DUTS [33] in our model training.

To fit the DUTS dataset to the Co-SOD task, [48] divided its images into different groups based on the categories of salient objects, obtaining the DUTS class dataset, which contains 8250 images of 291 groups. However, each image in this dataset only contains target salient objects without distractions. To this end, [48] synthesized images for their model training by using a jigsaw strategy. This method splices each image of the target class with an image from other classes. Although this strategy can simulate the distraction from extraneous salient objects in Co-SOD, it still has drawbacks that the splicing results are unnatural and objects will have large distortions when the synthesized images are resized to fixed shapes for network training.

Instead, we propose a copy and blend synthesis strategy based on poisson blending [25]. For each image of the target class, we randomly select an image from other classes and then copy and blend its salient object on the target image background as the distraction to generate the synthesized image. However, for images synthesized in such a normal way, the target objects are usually more salient than the copied extraneous objects. As a result, the trained models easily downgrade to only learn to detect salient objects instead of co-salient objects. To tackle this issue, we also propose a reverse synthesis strategy to copy and blend target objects on the backgrounds of extraneous images using the same aforementioned synthesis method. Finally, we combine both normal and reverse strategies to train our model.

Compared with [48], our proposed method can achieve more natural synthesis results and preserve reasonable shapes for objects, hence is more suitable for training Co-SOD models. Figure 4 shows some synthesized examples generated by our proposed method.

5. Experiments

5.1. Datasets and Evaluation Metrics

We evaluate our proposed method on four co-saliency benchmark datasets as follows. **MSRC** [37] is collected for recognizing objects and we follow [43, 7] to select 233 images of 7 groups from MSRC for evaluation. **CoSal2015** [42] and **CoSOD3k** [6] are two large-scale datasets containing 2015 images of 50 groups and 3316 images of 160 groups, respectively. **CoCA** [48] is the latest and most challenging dataset which contains 1295 images of 80 groups. Different from other datasets, each image in CoCA contains at least one extraneous salient object, hence being more suitable for real-world applications and evaluating the performance of Co-SOD methods.

We adopt four widely-used evaluation metrics to compare our proposed method with state-of-the-art methods. Maximum F-measure (maxF) considers both precision and recall for co-saliency maps binarized by an optimal threshold. Structure-measure S_m [3] considers object-aware and region-aware structural similarities. Enhanced-alignment measure E_ξ [4] considers both global information and local details. Mean Absolute Error (MAE) computes the average absolute per-pixel difference between the predicted co-saliency maps and ground truth.

5.2. Implementation Details

In our data synthesis strategy, for each original image in DUTS class, we generate three synthesized images using the normal strategy and another three using the reverse strategy.

We follow [21] to conduct data augmentation and use 256×256 as the training and testing size. We employ the cross-entropy loss as the training loss and deploy deep supervision for each decoder module. Stochastic gradient descent is used as our optimization algorithm. We select at most 14 images from each group as each minibatch and set the total iteration step to 40,000. The initial learning rate is set to 0.01 and divided by 10 at the $20,000^{th}$ and the $30,000^{th}$ iterations, respectively. Our code is implemented using Pytorch [24].

Table 1. Quantitative results of different settings of our proposed model. "VAK" and "VCK" mean vanilla adaptive kernels and the vanilla common kernel, respectively. "LAK" and "LCK" represent large adaptive kernels and the large common kernel. "ML" means adopting CADC at multiple decoder levels.

Settings	CoCA			
	$S_m \uparrow$	maxF $\uparrow$	$E_\xi \uparrow$	MAE $\downarrow$
Baseline	0.633	0.451	0.707	0.165
+VAK	0.657	0.495	0.729	0.153
+VCK	0.655	0.497	0.711	0.151
+LAK	0.661	0.508	0.735	0.146
+LCK	0.659	0.498	0.722	0.147
+LAK+LCK	0.665	0.511	0.731	0.144
+LAK+LCK+ML	**0.681**	**0.548**	**0.744**	**0.132**

Table 2. Quantitative results of using different training strategies.

Train strategy	CoCA			
	$S_m \uparrow$	maxF $\uparrow$	$E_\xi \uparrow$	MAE $\downarrow$
COCO-sub	0.628	0.467	0.707	0.171
+DUTS class [48]	0.645	0.494	0.720	0.165
+jigsaw strategy [48]	0.669	0.537	0.740	0.149
+normal strategy	0.653	0.504	0.725	0.157
+reverse strategy	0.653	0.510	0.735	0.155
+bidirectional strategy	**0.681**	**0.548**	**0.744**	**0.132**

5.3. Ablation Study

We conduct ablation studies on the most challenging and the latest Co-SOD dataset CoCA [48].

Effectiveness of CADC. The first row in Table 1 denotes our baseline model, *i.e*., employing UNet and DASPP with five simple decoders. This model degenerates to a pure SOD model without considering consensus information among images. Next, we separately use vanilla adaptive kernels (+VAK) and the vanilla common kernel (+VCK) in the baseline to incorporate consensus summarization. It can be seen that vanilla kernels obviously gain improvements when compared with the baseline. Furthermore, by adopting the efficient large dynamic kernels, *i.e*., large adaptive kernels (+LAK) and the large common kernel (+LCK), the model performance can be further improved when compared with using vanilla kernels. Figure 5 also indicates that larger kernels can better search co-occurring objects while vanilla kernels can be easily interfered with by distraction objects or miss to completely segment the whole object. Combining these two kinds of kernels (+LAK+LCK) can bring more performance gains, indicating that consensus object searching can be better performed in a supplementary way. Figure 6 also indicates adaptive kernels and common kernels can provide supplemental information.

Furthermore, we conduct hierarchical object searching on multiple levels (+LAK+LCK+ML), *i.e*., in the first four decoders. We can find that using dynamic convolution on multi-level feature maps can significantly bringing performance improvements. Hence, we use this setting as our final CADC network.

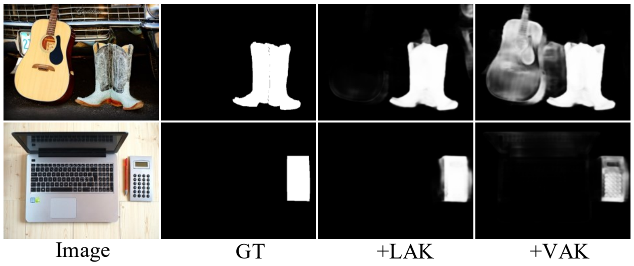

Figure 5. Visual comparison between "+LAK" and "+VAK".

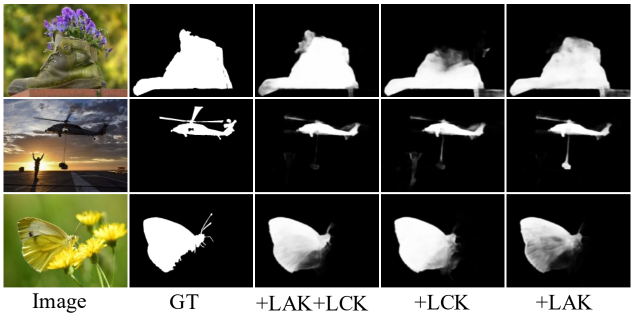

Figure 6. Visual comparison among "+LAK", "+LCK", and "+LAK+LCK".

Effectiveness of our data synthesis strategy. Table 2 shows the comparison results of training our model on different data. We first train our model on the COCO subset, *i.e*., COCO-sub. Then, we respectively add the original DUTS class dataset, the synthesized images using the jigsaw strategy [48], synthesizing only using our normal synthesis strategy, synthesizing only using our reverse synthesis strategy, and using our bidirectional synthesis strategy (normal and reverse). The results show that adding original DUTS class images can bring performance gains compared to only using the COCO-sub dataset, indicating the supplementary of saliency attributes is necessary. Furthermore, only using our normal or reverse strategy can obtain slightly better results than using the original DUTS class data. However, using both of them can lead to large performance gains and outperform the jigsaw strategy. Hence, our normal and reverse synthesis strategies provide complementary cues to each other and both of them are indispensable for effective model training.

5.4. Comparison with State-of-the-Art Methods

We compare our proposed model with other 11 state-of-the-art methods, *i.e*., CBCS [7], DIM [41], CODW [42], MIL [45], IML [27], SP-MIL [44], GONet [13], CSMG [46], GCAGC [47], GICD [48] , and ICNet [16].

We illustrate the quantitative comparison results in table 3. Generally, our model achieves the best performance on all four datasets. On the most challenging dataset CoCA, our model brings 3.8% improvement in terms of maxF compared with the second-best method. We also show quali-

Table 3. Quantitative comparison of our proposed model with other 11 SOTA Co-SOD methods on 4 benchmark datasets. **Red** and **blue** denote the best and the second-best results, respectively. '-' indicates the code or result is not available.

Dataset	Metric	CBCS	DIM	CODW	MIL	IML	SP-MIL	GONet	CSMG	GCAGC	GICD	ICNet	Ours
		[7]	[41]	[42]	[45]	[27]	[44]	[13]	[46]	[47]	[48]	[16]	
	S_m ↑	0.526	-	-	-	-	-	-	0.632	-	**0.658**	0.651	**0.681**
CoCA	maxF↑	0.315	-	-	-	-	-	-	0.508	-	**0.510**	0.506	**0.548**
	E_ξ ↑	0.638	-	-	-	-	-	-	**0.735**	-	0.712	0.698	**0.744**
[48]	MAE↓	0.175	-	-	-	-	-	-	**0.124**	-	**0.125**	0.148	0.132
	S_m ↑	0.528	0.559	-	-	0.720	-	-	0.711	-	0.778	**0.780**	**0.801**
CoSOD3k	maxF↑	0.466	0.495	-	-	0.652	-	-	0.709	-	**0.744**	**0.744**	**0.759**
	E_ξ ↑	0.637	0.662	-	-	0.773	-	-	0.804	-	0.831	**0.832**	**0.840**
[6]	MAE↓	0.228	0.327	-	-	0.164	-	-	0.157	-	**0.089**	0.097	**0.096**
	S_m ↑	0.544	0.592	0.648	0.673	-	-	0.751	0.774	0.822	0.842	**0.856**	**0.866**
CoSal2015	maxF↑	0.532	0.580	0.667	0.620	-	-	0.740	0.784	0.843	0.840	**0.855**	**0.862**
	E_ξ ↑	0.656	0.695	0.752	0.720	-	-	0.805	0.842	-	0.885	**0.900**	**0.906**
[42]	MAE↓	0.233	0.312	0.274	0.210	-	-	0.160	0.130	0.089	0.071	**0.058**	**0.064**
	S_m ↑	0.480	0.657	0.713	0.720	0.781	0.769	**0.795**	0.722	-	0.665	0.731	**0.821**
MSRC	maxF↑	0.630	0.705	0.784	0.768	0.840	0.824	0.846	**0.847**	-	0.692	0.805	**0.873**
	E_ξ ↑	0.676	0.725	0.820	0.8	0.856	0.855	**0.863**	0.859	-	0.726	0.822	**0.895**
[37]	MAE↓	0.314	0.309	0.264	0.216	0.174	0.218	0.179	0.190	-	0.196	**0.160**	**0.115**

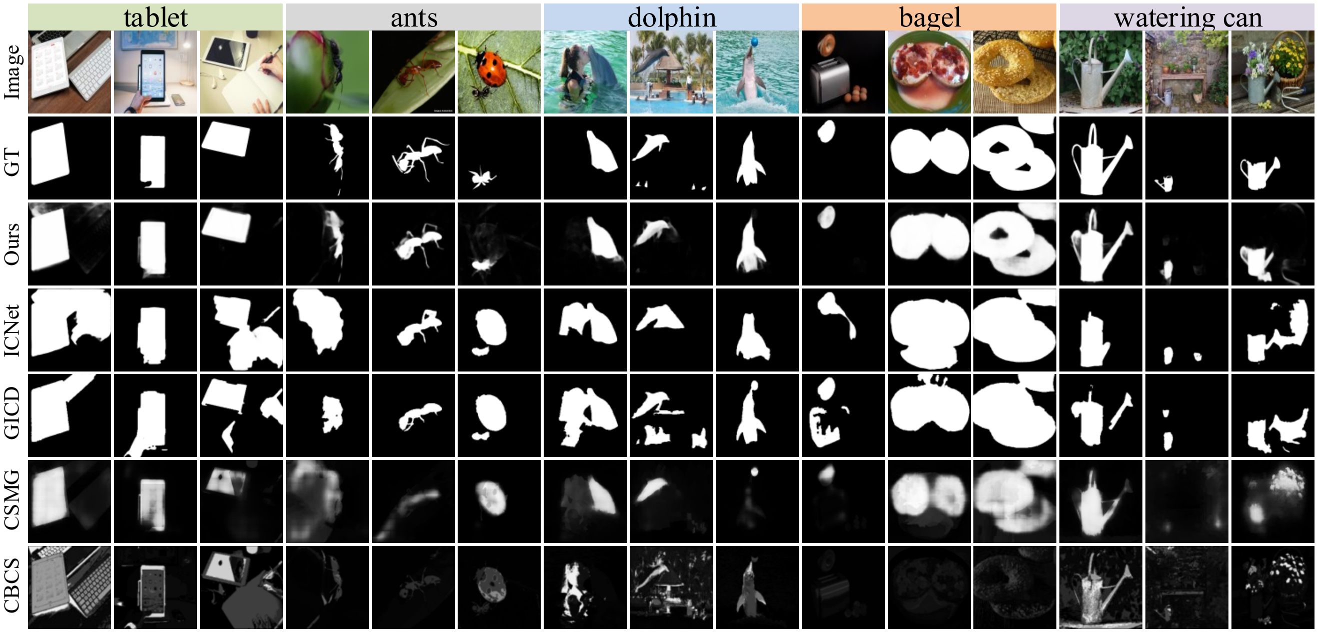

Figure 7. Qualitative comparisons of our proposed model with other state-of-the-art methods.

tative comparison results in Figure 7. It can be seen that our model can better search and segment the co-occurring salient objects in many challenging scenes while other methods often are disturbed by other extraneous salient objects. Specifically, for the ants class, our model can accurately search the targets which are similar to the background while other methods either lost the targets or be interfered with by other salient objects.

6. Conclusion

In this paper, we propose a consensus-aware dynamic convolution model to explicitly perform the "summarize and search" process for co-saliency detection. Two types of efficient large dynamic kernels are constructed in a supplementary way to capture image-specific consensus object cues and the group-wise common knowledge, respectively. We hierarchically search the co-salient objects by performing the dynamic convolution operation at multiple levels. We also present a new data synthesis method to effectively mimic the distraction of extraneous objects in the real world. Extensive experimental results demonstrate the effectiveness of our proposed method.

Acknowledgments: This work was supported in part by the National Key R&D Program of China under Grant 2018YFB1402600, the National Science Foundation of China under Grant 62027813, 62036005, U20B2065, U20B2068.

References

[1] Qi Cai, Yingwei Pan, Ting Yao, Chenggang Yan, and Tao Mei. Memory matching networks for one-shot image recognition. In *CVPR*, pages 4080–4088, 2018.

[2] Yue Cao, Jiarui Xu, Stephen Lin, Fangyun Wei, and Han Hu. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *ICCV Workshops*, 2019.

[3] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *ICCV*, pages 4548–4557, 2017.

[4] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment Measure for Binary Foreground Map Evaluation. In *IJCAI*, pages 698–704, 2018.

[5] Deng-Ping Fan, Tengpeng Li, Zheng Lin, Ge-Peng Ji, Dingwen Zhang, Ming-Ming Cheng, Huazhu Fu, and Jianbing Shen. Re-thinking co-salient object detection. *TPAMI*, 2021.

[6] Deng-Ping Fan, Zheng Lin, Ge-Peng Ji, Dingwen Zhang, Huazhu Fu, and Ming-Ming Cheng. Taking a deeper look at co-salient object detection. In *CVPR*, pages 2919–2929, 2020.

[7] Huazhu Fu, Xiaochun Cao, and Zhuowen Tu. Cluster-based co-saliency detection. *TIP*, 22(10):3766–3778, 2013.

[8] Guangshuai Gao, Wenting Zhao, Qingjie Liu, and Yunhong Wang. Co-saliency detection with co-attention fully convolutional network. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(3):877–889, 2020.

[9] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, pages 4367–4375, 2018.

[10] Junwei Han, Gong Cheng, Zhenpeng Li, and Dingwen Zhang. A unified metric learning-based framework for co-saliency detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(10):2473–2483, 2017.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, pages 1026–1034, 2015.

[12] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[13] Kuang-Jui Hsu, Chung-Chi Tsai, Yen-Yu Lin, Xiaoning Qian, and Yung-Yu Chuang. Unsupervised cnn-based co-saliency detection with graphical optimization. In *ECCV*, pages 485–501, 2018.

[14] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456, 2015.

[15] Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc V Gool. Dynamic filter networks. In *NIPS*, pages 667–675, 2016.

[16] Wen-Da Jin, Jun Xu, Ming-Ming Cheng, Yi Zhang, and Wei Guo. Icnet: Intra-saliency correlation network for co-saliency detection. *NIPS*, 2020.

[17] Bo Li, Zhengxing Sun, Lv Tang, Yunhan Sun, and Jinlong Shi. Detecting robust co-saliency with recurrent co-attention neural network. In *IJCAI*, pages 818–825, 2019.

[18] Hongliang Li and King Ngi Ngan. A co-saliency model of image pairs. *TIP*, 20(12):3365–3375, 2011.

[19] Min Li, Shizhong Dong, Kun Zhang, Zhifan Gao, Xi Wu, Heye Zhang, Guang Yang, and Shuo Li. Deep learning intra-image and inter-images features for co-saliency detection. In *BMVC*, volume 291, 2018.

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014.

[21] Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Learning pixel-wise contextual attention for saliency detection. In *CVPR*, pages 3089–3098, 2018.

[22] Nian Liu, Ni Zhang, and Junwei Han. Learning selective self-mutual attention for rgb-d saliency detection. In *CVPR*, pages 13756–13765, 2020.

[23] Youwei Pang, Lihe Zhang, Xiaoqi Zhao, and Huchuan Lu. Hierarchical dynamic filtering network for rgb-d salient object detection. In *ECCV*, pages 235–252, 2020.

[24] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NIPS*, 32:8026–8037, 2019.

[25] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In *ACM SIGGRAPH 2003 Papers*, pages 313–318. 2003.

[26] Lu Qi, Yi Wang, Yukang Chen, Ying-Cong Chen, Xiangyu Zhang, Jian Sun, and Jiaya Jia. Pointins: Point-based instance segmentation. *TPAMI*, 2021.

[27] Jingru Ren, Zhi Liu, Xiaofei Zhou, Cong Bai, and Guangling Sun. Co-saliency detection via integration of multi-layer convolutional features and inter-image propagation. *Neurocomputing*, 371:137–146, 2020.

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241, 2015.

[29] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[30] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *ECCV*, pages 282–298, 2020.

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

[32] Chong Wang, Zheng-Jun Zha, Dong Liu, and Hongtao Xie. Robust deep co-saliency detection with group semantic. In *AAAI*, volume 33, pages 8917–8924, 2019.

[33] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *CVPR*, pages 136–145, 2017.

[34] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, pages 7794–7803, 2018.

[35] Lina Wei, Shanshan Zhao, Omar El Farouk Bourahla, Xi Li, and Fei Wu. Group-wise deep co-saliency detection. In *IJCAI*, pages 3041–3047, 2017.

[36] Lina Wei, Shanshan Zhao, Omar El Farouk Bourahla, Xi Li, Fei Wu, and Yueting Zhuang. Deep group-wise fully convolutional network for co-saliency detection with graph propagation. *TIP*, 28(10):5052–5063, 2019.

[37] John Winn, Antonio Criminisi, and Thomas Minka. Object categorization by learned universal visual dictionary. In *ICCV*, volume 2, pages 1800–1807, 2005.

[38] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In *CVPR*, pages 3684–3692, 2018.

[39] Xiwen Yao, Junwei Han, Dingwen Zhang, and Feiping Nie. Revisiting co-saliency detection: A novel approach based on two-stage multi-view spectral rotation co-clustering. *TIP*, 26(7):3196–3209, 2017.

[40] Zheng-Jun Zha, Chong Wang, Dong Liu, Hongtao Xie, and Yongdong Zhang. Robust deep co-saliency detection with group semantic and pyramid attention. *TNNLS*, 31(7):2398–2408, 2020.

[41] Dingwen Zhang, Junwei Han, Jungong Han, and Ling Shao. Cosaliency detection based on intrasaliency prior transfer and deep intersaliency mining. *TNNLS*, 27(6):1163–1176, 2015.

[42] Dingwen Zhang, Junwei Han, Chao Li, and Jingdong Wang. Co-saliency detection via looking deep and wide. In *CVPR*, pages 2994–3002, 2015.

[43] Dingwen Zhang, Junwei Han, Chao Li, Jingdong Wang, and Xuelong Li. Detection of co-salient objects by looking deep and wide. *IJCV*, 120(2):215–232, 2016.

[44] Dingwen Zhang, Deyu Meng, and Junwei Han. Co-saliency detection via a self-paced multiple-instance learning framework. *TPAMI*, 39(5):865–878, 2016.

[45] Dingwen Zhang, Deyu Meng, Chao Li, Lu Jiang, Qian Zhao, and Junwei Han. A self-paced multiple-instance learning framework for co-saliency detection. In *ICCV*, pages 594–602, 2015.

[46] Kaihua Zhang, Tengpeng Li, Bo Liu, and Qingshan Liu. Co-saliency detection via mask-guided fully convolutional networks with multi-scale label smoothing. In *CVPR*, pages 3095–3104, 2019.

[47] Kaihua Zhang, Tengpeng Li, Shiwen Shen, Bo Liu, Jin Chen, and Qingshan Liu. Adaptive graph convolutional network with attention graph clustering for co-saliency detection. In *CVPR*, pages 9050–9059, 2020.

[48] Zhao Zhang, Wenda Jin, Jun Xu, and Ming-Ming Cheng. Gradient-induced co-saliency detection. In *ECCV*, pages 455–472, 2020.

[49] Jia-Xing Zhao, Jiang-Jiang Liu, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. Egnet: Edge guidance network for salient object detection. In *ICCV*, pages 8779–8788, 2019.