

CONDA: Condensed Deep Association Learning for Co-Salient Object Detection

Long Li¹, Nian Liu^{2,*}, Dingwen Zhang¹, Zhongyu Li³, Salman Khan^{2,4},
Rao Anwer², Hisham Cholakkal², Junwei Han^{1,*}, and Fahad Shahbaz
Khan^{2,5}

¹ Northwestern Polytechnical University

² Mohamed bin Zayed University of Artificial Intelligence

³ Xi'an Jiaotong University

⁴ Australian National University

⁵ CVL, Linköping University

Abstract. Inter-image association modeling is crucial for co-salient object detection. Despite satisfactory performance, previous methods still have limitations on sufficient inter-image association modeling. Because most of them focus on image feature optimization under the guidance of heuristically calculated raw inter-image associations. They directly rely on raw associations which are not reliable in complex scenarios, and their image feature optimization approach is not explicit for inter-image association modeling. To alleviate these limitations, this paper proposes a deep association learning strategy that deploys deep networks on raw associations to explicitly transform them into deep association features. Specifically, we first create hyperassociations to collect dense pixel-pair-wise raw associations and then deploys deep aggregation networks on them. We design a progressive association generation module for this purpose with additional enhancement of the hyperassociation calculation. More importantly, we propose a correspondence-induced association condensation module that introduces a pretext task, *i.e.* semantic correspondence estimation, to condense the hyperassociations for computational burden reduction and noise elimination. We also design an object-aware cycle consistency loss for high-quality correspondence estimations. Experimental results in three benchmark datasets demonstrate the remarkable effectiveness of our proposed method with various training settings. The code is available at: <https://github.com/dragonlee258079/CONDA>.

Keywords: Co-salient Object Detection · Deep Association Learning

1 Introduction

Co-Salient Object Detection (CoSOD) aims to segment salient objects that appear commonly across a group of related images. Compared to traditional Salient Object Detection (SOD) [18–22, 25, 45, 46, 48], CoSOD is a more challenging task because it requires sufficient inter-image association modeling [9].

* Corresponding authors: {liunian228, junweihan2010}@gmail.com

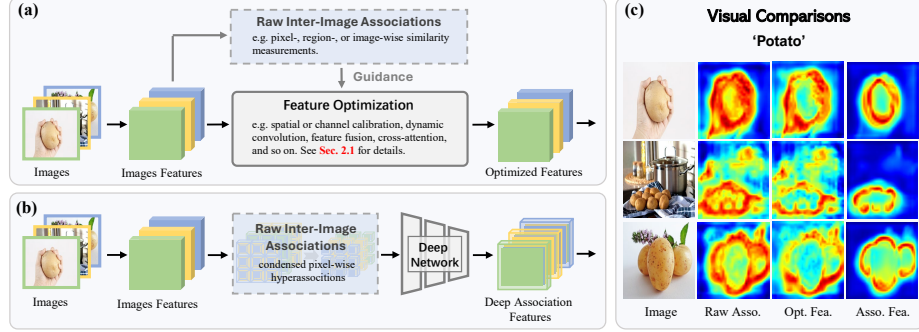


Fig. 1: Difference of raw-association-based image feature optimization strategy (a) and our proposed deep association learning strategy (b). Our deep association learning deploys deep learning networks on raw associations to achieve deep association features. We also present visual samples of our calculated raw associations (Raw Asso.), optimized image feature (Opt. Fea.), and our generated deep association features (Asso. Fea.) in (c).

Recently, many advanced works [8, 12, 15, 32, 37, 40, 42, 43, 47] have emerged and achieved impressive performance. These methods first use related image features to acquire raw inter-image associations (also called consensus representations), and then leverage them as guidance to optimize each image feature, as shown in Figure 1 (a). This approach enables the final image feature to implicitly capture inter-image cues, thereby achieving the purpose of inter-image association modeling. However, we find this *raw-association-based feature optimization* strategy still has two limitations: i) they directly rely on raw associations, which are acquired in heuristic manners, such as pixel-wise [8, 32, 40, 47], region-wise [15, 42], or image-wise [12] similarity measurements. Although high-quality raw associations can be derived from high-level semantic information in image features, their revelation of common saliency regions still relies on similarity measures, which are unreliable when encountering complex scenarios, such as significant differences between co-salient objects or high foreground-background similarity. ii) the primary focus of building their deep models is on optimizing image features. Compared to directly modeling association relationship, image feature optimization is not an explicit approach for inter-image association modeling and will increase the learning difficulty.

To alleviate these limitations, we propose a *deep association learning* strategy for CoSOD, as shown in Figure 1 (b). Instead of directly using raw associations to optimize image features, we deploy deep networks on raw associations to learn deep association features. This is a more explicit strategy for inter-image association modeling. Moreover, our deep association features can capture high-level inter-image association knowledge, making them more robust in complex scenarios than raw associations, as shown in Figure 1 (c). Technically, we start by collecting all pixel-wise raw associations across the entire image group as hyperassociations. Then, we propose a Progressive Association Generation (PAG) module to transform hyperassociations into deep association features. PAG progressively generates association features on varying scales, allowing us to use

previous association features to enhance the hyperassociation calculation at the next scale, thereby improving the association quality from the very beginning.

Although deep association learning strategy allows more sufficient inter-image association modeling, it significantly increases the computational burden and reduces the practicality of this approach. Additionally, this study finds that it is not necessary to utilize all pixel associations to generate deep association features. In fact, there are even some noisy pixels that negatively impact the quality of the deep association features. Therefore, we propose a method based on Correspondence-induced Association Condensation (CAC) to condense the original full-pixel hyperassociations. This not only alleviates the computational burden but also further enhances the quality of deep association features.

Specifically, CAC performs the condensation operation by selectively associating pixels that have semantic correspondence in other images, as well as their surrounding contextual pixels, thereby creating lightweight yet more accurate hyperassociations. Here, we introduce a pretext task, *i.e.* semantic correspondence estimation, into the CoSOD, not only improving the model performance but also delving deeper into the essence of CoSOD. Co-salient objects inherently possess *object-level* semantic correspondence. However, in this paper, we aim to further explore the finer *pixel-level* correspondence. Although highly accurate correspondence estimation remains a challenge, we believe it will pave a new way for CoSOD research. We also provide an object-aware cycle consistency (OCC) loss to aid in learning correspondences within co-salient pixels.

In summary, the contributions of this paper are as follows:

- For the first time, we introduce a *deep association learning* approach for CoSOD, applying deep networks to transform raw associations into deep association features for sufficient inter-image association modeling. Specifically, we develop a **COND**ensed **Deep Association** (CONDA) learning model.
- We propose a PAG module to progressively generate deep association features. It enhances image features with previous association features to improve hyperassociation calculation.
- We introduce semantic correspondence estimation into the CoSOD task to condense the original hyperassociation for alleviating the computational burden and further improving the performance. We also propose an OCC loss for effective correspondence estimation.
- Experimental results demonstrate that our model achieves significantly improved state-of-the-art performance on three benchmark datasets across different training settings.

2 Related Work

2.1 Co-Salient Object Detection

Recently, there has been a surge of excellent methods for CoSOD [8, 12, 15, 32, 37, 40, 42, 43, 47]. These methods initially acquire raw inter-image associations with related image features and then utilize them to optimize each image feature.

Most methods generate pixel-wise [8, 32, 40, 47], region-wise [15, 42], and image-wise [12] raw associations through similarity-based manners, *e.g.* inner product calculations between image features. Even for transformer-based methods [15, 32], they also rely on inner product calculation to produce attention maps [35] as raw associations. The image feature optimization manners include spatial or channel calibration [8, 12, 47], dynamic convolution [40, 42], feature fusion [15, 40], and cross-attention [15, 32], etc. However, they lack the learning of high-level association knowledge and heavily focus on optimized image features. Unlike them, this paper proposes a new research direction that deploys deep networks on associations to achieve deep association features for CoSOD.

2.2 Inter-image Relation Modeling

Apart from CoSOD, other tasks necessitating the consideration of inter-image relationships, such as few-shot segmentation [24, 26, 27, 38], stereo matching [2, 39], video semantic segmentation [33], etc. These tasks have made significant advancements recently by effectively modeling inter-image relations. Most of these methods [2, 4, 11, 26, 38, 39] initially create cost volumes to capture dense inter-image pixel-wise similarities and subsequently use various modules to convert these hyper volumes into task-specific features. Our approach distinguishes itself from prior methods in three aspects. Firstly, most of them create 4D cost volumes between two images while we create 6D hyperassociations between all related images. Secondly, they calculate hyper volumes based on original image features, whereas We propose PAG to progressively enhance image features for better hyperassociation calculation. Last and most importantly, they rely on the full-pixel cost volume. We consider the condensation of hyperassociations using semantic correspondences to eliminate noisy pixel associations.

2.3 Semantic Correspondence Estimation

Semantic correspondence estimation [23] aims to establish reliable pixel correspondences between different instances of the same object category. Most works performed this task using fully supervised training [13, 23]. Some recent works have utilized unsupervised learning with photometric, forward-backward consistency, and warp-supervision losses [29, 34, 44]. However, they implement these losses on the entire image, where background pixels may affect the performance. In this paper, we introduce this task to condense hyperassociations for Co-SOD and tailor the cycle consistency loss by only applying it to co-salient pixels, hence effectively avoiding the influence of background and extraneous objects.

3 Proposed Method

As shown in Figure 2, CONDA integrates the deep association learning process into an FPN framework. Specifically, given a group of related images $\{\mathbf{I}_i\}_{i=1}^N$, we first input them into a VGG-16 [31] backbone to collect its intermediate features

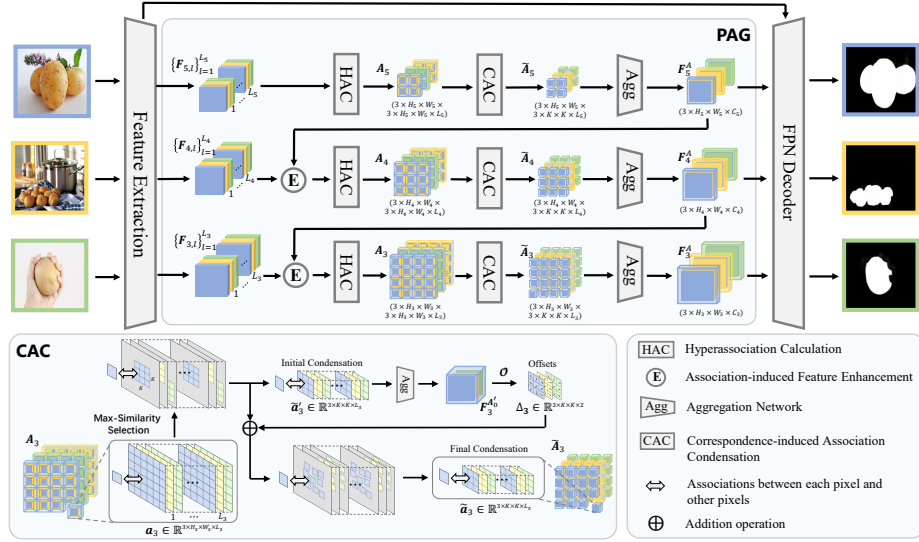


Fig. 2: Overall flowchart of our CONDA model. Specifically, CONDA first utilizes the image features to calculate hyperassociations. Then, the full-pixel hyperassociations are condensed by CAC and fed into the aggregation networks to achieve deep association features. These features are then used in the FPN decoder process for the final prediction. To be concise, only three related images are shown.

for PAG and FPN decoding. In detail, we collect all features of the last three stages for PAG and the last feature of each stage for the FPN decoder as follows:

$$\begin{aligned}\mathbb{F}^P &= \{\mathbf{F}_{s,l} \mid s \in \{3, \dots, 5\}, l \in \{1, \dots, L_s\}\}, \\ \mathbb{F}^D &= \{\mathbf{F}_{s,L_s} \mid s \in \{1, \dots, 5\}\},\end{aligned}\quad (1)$$

where $\mathbb{F}^P$ and $\mathbb{F}^D$ are the feature collections for PAG and FPN decoder, respectively. $\mathbf{F}_{s,l} \in \mathbb{R}^{N \times H_s \times W_s \times C_s}$ represents the VGG feature of the l -th layer in the s -th stage. There are in total L_s layers in the s -th stage. H_s , W_s and C_s represent the height, width, and channel of the s -th stage, respectively.

Then, we input $\mathbb{F}^P$ into PAG to calculate hyperassociations and generate deep association features $\{\mathbf{F}_s^A \in \mathbb{R}^{N \times H_s \times W_s \times C_s}\}_{s=3}^5$. Finally, these association features will be fused with $\mathbb{F}^D$ for the FPN decoder process, formulated as:

$$\begin{aligned}\mathbf{F}_{s,L_s} &= \mathbf{F}_{s,L_s} + \phi(\mathbf{F}_s^A), \quad s = 3, \dots, 5, \\ \mathbf{F} &= \text{Decoder}(\{\mathbf{F}_{s,L_s}\}_{s=1}^5),\end{aligned}\quad (2)$$

where ϕ is a convolution layer. $\mathbf{F} \in \mathbb{R}^{N \times H \times W \times C}$ is the final feature for the final co-saliency prediction. We adopt BCE and IoU losses for supervision.

The rest of this section will introduce PAG with full-pixel hyperassociation and the condensation of hyperassociations by plugging the Correspondence-induced Association Condensation (CAC) module into PAG.

3.1 Progressive Association Generation

Our deep association learning involves two steps: 1) acquiring the raw hyperassociations $\{\mathbf{A}_s\}_{s=3}^5$ from three stages; 2) employing aggregation networks on $\{\mathbf{A}_s\}_{s=3}^5$ to obtain association features $\{\mathbf{F}_s^A\}_{s=3}^5$.

Earlier methods [4, 10, 30] directly utilize the original backbone features, *i.e.* $\mathbb{F}^P$, to calculate inter-image interactions, such as the so-called cost volume. We argue that hyperassociations derived straight from backbone features might obstruct further improvement of deep association learning, given that the current backbone is pre-trained without any consideration for inter-image associations.

To alleviate this problem, we propose the PAG module to progressively generate pyramid association features so that we can utilize the high-level association feature, *e.g.* $\mathbf{F}_{s+1}^A$ from the $(s+1)$ -th stage, to enhance the neighbouring low-level VGG features in $\mathbb{F}^P$, *e.g.* $\mathbf{F}_{s,l}$, for attaining association-enhanced features $\hat{\mathbf{F}}_{s,l}$, based on which we can calculate high-quality hyperassociations $\mathbf{A}_s$ in the s -th stage. After that, we execute the subsequent aggregation networks on $\mathbf{A}_s$ to achieve association features $\mathbf{F}_s^A$, which will continue to enhance the VGG features of the next stage and carry out progressive association generation. The whole process of our PAG can be formulated as follows:

$$\begin{aligned} \mathbf{A}_s &= \text{HAC}(\{\hat{\mathbf{F}}_{s,l}\}_{l=1}^{L_s}), \\ \mathbf{F}_s^A &= \text{Agg}(\mathbf{A}_s), \\ \{\hat{\mathbf{F}}_{s-1,l}\}_{l=1}^{L_{s-1}} &= \text{Enh}(\{\mathbf{F}_{s-1,l}\}_{l=1}^{L_{s-1}}; \mathbf{F}_s^A), \end{aligned} \quad (3)$$

where s ranges from 5 to 3 and $\{\hat{\mathbf{F}}_{5,l}\}_{l=1}^{L_5} = \{\mathbf{F}_{5,l}\}_{l=1}^{L_5}$. The HAC, Agg, and Enh represent the hyperassociation calculation, aggregation network, and association-induced feature enhancement, respectively. Next, we explain them in detail.

Hyperassociation Calculation. For each stage, we first compute the raw associations at each layer using the inner product between l -2 normalized association-enhanced features of N related images. After that, we stack the raw associations of all layers to form the final hyperassociation of this stage. The hyperassociation for the s -th stage, *i.e.* $\mathbf{A}_s \in \mathbb{R}^{N \times H_s \times W_s \times N \times H_s \times W_s \times L_s}$, can be calculated via:

$$\begin{aligned} \mathbf{A}_s &= \text{HAC}(\{\hat{\mathbf{F}}_{s,l}\}_{l=1}^{L_s}), \\ &= \text{Stack}\left(\left\{\text{ReLU}\left(\frac{\hat{\mathbf{F}}_{s,l} \cdot \hat{\mathbf{F}}_{s,l}^\top}{\|\hat{\mathbf{F}}_{s,l}\| \|\hat{\mathbf{F}}_{s,l}^\top\|}\right)\right\}_{l=1}^{L_s}\right), \end{aligned} \quad (4)$$

where $\hat{\mathbf{F}}_{s,l} \cdot \hat{\mathbf{F}}_{s,l}^\top \in \mathbb{R}^{N \times H_s \times W_s \times N \times H_s \times W_s}$. The $\top$ indicates transposing the last dimension and the first three dimensions. $\|\cdot\|$ represents the l -2 norm. We employ ReLU to suppress noisy association values.

Aggregation Network. The raw hyperassociation $\mathbf{A}_s \in \mathbb{R}^{N \times H_s \times W_s \times N \times H_s \times W_s \times L_s}$ is a hypercube with a nested structure, where each pixel position is characterized by a 4D tensor $(N \times H_s \times W_s \times L_s)$. Each 4D tensor documents the associations of the respective pixel with all other pixels in N related images. For

clarity, we designate the first and second $N \times H_s \times W_s$ dimensions in $\mathbf{A}_s$ as the source and target dimensions, respectively. Although these 4D tensors are crucial for exploring the consensus information for co-saliency detection, they essentially comprise pixel-to-pixel similarity values, seen in (4), which may be suboptimal and unreliable in complex scenarios. Therefore, we propose *using deep networks* to transform these pixel-wise similarities into deep association features with contextual and high-order association knowledge. This has never been explored in previous CoSOD methods. This is implemented via context aggregations on $\mathbf{A}_s$ to squeeze these 4D tensors as C_s -dimensional vectors, formulated as:

$$N \times H_s \times W_s \times N \times H_s \times W_s \times L_s \rightarrow N \times H_s \times W_s \times C_s. \quad (5)$$

In detail, we first deploy several association aggregation layers on $\mathbf{A}_s$ to progressively aggregate context information, enlarging L_s as C_s , and eliminate the target $H_s \times W_s$ dimension in 4D tensors. Each aggregation layer consists of 2D convolution layers and a downsampling operation. Specifically, focusing on the first aggregation layer for technical explanation, we first aggregate context information in the target $H_s \times W_s$ dimension by applying a 2D convolution layer on all 4D tensors. The operations on the 4D tensor at pixel position (h_i, w_i) in image I_i can be formulated as:

$$\mathbf{A}_s^1(i, h_i, w_i, j, :, :, :) = \mathcal{C}_1(\mathbf{A}_s(i, h_i, w_i, j, :, :, :)), \quad j = 1, \dots, N, \quad (6)$$

where $\mathcal{C}_1$ is a 3×3 2D convolution layer. Here, j is the index of other related images in the 4D tensor, and $\mathbf{A}_s(i, h_i, w_i, j, :, :, :) \in \mathbb{R}^{H_s \times W_s \times L_s}$ illustrates the associations between the pixel (h_i, w_i) in I_i and all pixels in image I_j . This interpretation applies to other similar symbols.

Then, a downsampling operation $\mathcal{D}$, *i.e.* bilinear interpolation, is applied on $\mathbf{A}_s^1$ to reduce the spatial dimension of the 4D tensor by a scaling factor:

$$\mathbf{A}_s^2(i, h_i, w_i, j, :, :, :) = \mathcal{D}(\mathbf{A}_s^1(i, h_i, w_i, j, :, :, :)), \quad (7)$$

where $\mathbf{A}_s^2(i, h_i, w_i, j, :, :, :) \in \mathbb{R}^{H'_s \times W'_s \times C'_s}$, H'_s and W'_s are downsampled height and width. C'_s is the channel number after convolution $\mathcal{C}_1$.

Finally, we also aggregate context information in the source $H_s \times W_s$ dimension. Specifically, we extract the values along the source dimension and channel dimension in $\mathbf{A}_s^2$ to form 4D tensors and apply a 2D convolution layer on them. For instance, $\mathbf{A}_s^2(:, :, :, j, h_j, w_j, :) \in \mathbb{R}^{N \times H_s \times W_s \times C'_s}$ is such a 4D tensor, where (h_j, w_j) is a pixel position in the target dimension. This can be formulated as:

$$\mathbf{A}_s^3(i, :, :, j, h_j, w_j, :) = \mathcal{C}_2(\mathbf{A}_s^2(i, :, :, j, h_j, w_j, :)), \quad i = 1, \dots, N, \quad (8)$$

where $\mathcal{C}_2$ is a 3×3 2D convolution layer. i is the related image index.

After several association aggregation layers, as shown in (6)-(8), we can achieve the aggregated association features with the target $H_s \times W_s$ dimension eliminated, denoted as $\mathbf{F}_s^{A'} \in \mathbb{R}^{N \times H_s \times W_s \times N \times C}$. Subsequently, we average $\mathbf{F}_s^{A'}$ on its second N dimension and obtain the final association features

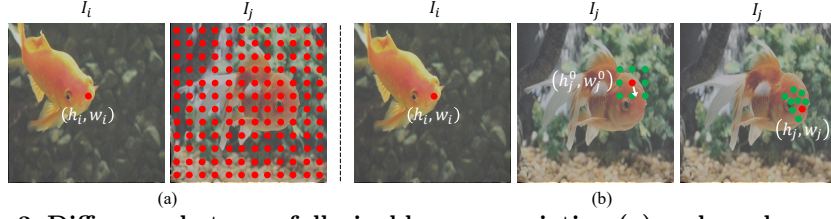


Fig. 3: Difference between full-pixel hyperassociation (a) and condensed hyperassociation (b). We provide an example of collecting the pixel associations from image I_j for a pixel (h_i, w_i) in image I_i . Full-pixel hyperassociation collects all pixel associations in I_j , while our condensed hyperassociation only collects the associations of its correspondence pixel (h_j, w_j) (red dot) and surrounding pixels (green dots). We first heuristically find an initial pixel (h_j^0, w_j^0) with a fixed surrounding window and then learn coordinate offsets to locate the optimized correspondence and surrounding pixels.

$\mathbf{F}_s^A \in \mathbb{R}^{N \times H_s \times W_s \times C}$, formulated as:

$$\mathbf{F}_s^A = \frac{1}{N} \sum_{j=1}^N \mathbf{F}_s^{A'}(:, :, :, j, :). \quad (9)$$

Association-induced Feature Enhancement. Once we have obtained association feature $\mathbf{F}_s^A$ of the s -th stage, we will use it to enhance the VGG feature of the $(s-1)$ -th stage, *i.e.* $\{\mathbf{F}_{s-1,l}\}_{l=1}^{L_{s-1}}$. Specifically, we upsample $\mathbf{F}_s^A$ to align the spatial size of features in the $(s-1)$ -th stage, and then add it to $\{\mathbf{F}_{s-1,l}\}_{l=1}^{L_{s-1}}$ followed by a convolution layer, formulated as:

$$\hat{\mathbf{F}}_{s-1,l} = \mathcal{C}_3(\mathbf{F}_{s-1,l} + \mathcal{U}(\mathbf{F}_s^A)), \quad (10)$$

where $\mathcal{C}_3$ and $\mathcal{U}$ represent a 2D convolution layer and the bilinear upsampling operation, respectively.

3.2 Correspondence-induced Association Condensation

Although PAG based on full-pixel hyperassociations can deliver satisfactory performance in CoSOD, it also introduces substantial computational overhead. Additionally, we argue that for each pixel in an image, it is unnecessary to gather its associations with all pixels of other related images to form hyperassociations. Some pixel associations may even impair the final performance, such as those between ambiguous regions. To this end, this subsection try to condense the original full-pixel hyperassociations to retain only informative pixel associations.

This subsection will focus on explaining the condensation of pixel associations of a pixel (*e.g.* (h_i, w_i) in I_i) to ones of the other images (*e.g.* image I_j), *i.e.* $\mathbf{A}_s(i, h_i, w_i, j, :, :, :) \in \mathbb{R}^{H_s \times W_s \times L_s}$, as shown in Figure 3. We will simplify the symbol $\mathbf{A}_s(i, h_i, w_i, j, :, :, :)$ as $\mathbf{a}_s^j$ in the subsequent text for convenience.

Specifically, CAC opts to select $K \times K$ ($K < H_s, K < W_s$) informative pixel associations from $\mathbf{a}_s^j$ to form its condensed representation, *i.e.* $\tilde{\mathbf{a}}_s^j \in \mathbb{R}^{K \times K \times L_s}$. Thus, the entire condensed hyperassocaition can be symbolized as $\mathbf{A}_s \in \mathbb{R}^{N \times H_s \times W_s \times N \times L_s}$.

$K \times K \times L_s$. To ensure the proper selection of $K \times K$ pixels, we introduce a pre-text task, *i.e.* semantic correspondence estimation [13, 23]. This allows us to first locate the corresponding pixel of (h_i, w_i) in image I_j , *i.e.* (h_j, w_j) , and then combine (h_j, w_j) with its surrounding pixels as the $K \times K$ pixel set. We design this approach based on the observation that the co-salient objects across N related images belong to the same semantic category, and the pixels within them should have semantic correspondences to each other, as shown in Figure 4. Therefore, the introduction of semantic correspondence in CAC not only improves the CoSOD performance but also delves deeper into the core nature of the CoSOD task. As far as we know, this is the first work to use semantic correspondence in the CoSOD task.

Correspondence Estimation. To estimate the correspondence pixel (h_j, w_j) in I_j for (h_i, w_i) , we first identify an initial pixel (h_j^0, w_j^0) via a heuristic approach. Subsequently, we produce a spatial offset to refine (h_j^0, w_j^0) into (h_j, w_j) . To achieve this purpose, all initial correspondence pixels should be utilized to form the initial condensed hyperassociations, with which the initial deep association features can be generated for spatial offset prediction.

Specifically, we pick out (h_j^0, w_j^0) that has the largest feature similarity value with (h_i, w_i) . Since we have calculated the feature similarities in $\mathbf{a}_s^j$, the (h_j^0, w_j^0) can be obtained as follows:

$$\begin{aligned} \bar{\mathbf{a}}_s^j &= \sum_{l=1}^{L_s} \mathbf{a}_s^j(:, :, l), \\ (h_j^0, w_j^0) &= \underset{x, y}{\operatorname{argmax}} (\bar{\mathbf{a}}_s^j(x, y)), \end{aligned} \quad (11)$$

where $\bar{\mathbf{a}}_s^j \in \mathbb{R}^{H_s \times W_s}$ is a similarity matrix obtained by eliminating the last dimension of $\mathbf{a}_s^j \in \mathbb{R}^{H_s \times W_s \times L_s}$ through summation. argmax returns the coordinate of the maximum value.

Next, we select $K \times K$ pixels within the square region centered around the initial pixels, *e.g.* (h_j^0, w_j^0) , to construct the initial condensed hyperassociation $\tilde{\mathbf{A}}_s'$. Then, we can feed it into the aggregation network, described in Sec 3.1, and achieve the initial aggregated association features $\mathbf{F}_s^{A'} \in \mathbb{R}^{N \times H_s \times W_s \times N \times C_s}$. It can be regarded as the association features of each pixel to N other related images. We utilize the feature of (h_i, w_i) to I_j , *i.e.* $\mathbf{F}_s^{A'}(i, h_i, w_i, j, :) \in \mathbb{R}^{C_s}$, to predict the offsets for (h_j, w_j) via a linear layer, formulated as:

$$\Delta_s^j = \mathcal{O}(\mathbf{F}_s^{A'}(i, h_i, w_i, j, :)), \quad (12)$$

where $\mathcal{O}$ is a linear layer for offset generation. $\Delta_s^j \in \mathbb{R}^{K \times K \times 2}$ consists of $K \times K$ offsets, besides the center offset $\Delta_s^j(k_c, k_c, :)$ for correspondence estimation, *i.e.* refining (h_j^0, w_j^0) as (h_j, w_j) , we also generate other offsets for surrounding pixel selection. Thus, the corresponding pixel (h_j, w_j) can be obtained by adding the offset to the initial pixel (h_j^0, w_j^0) , formulated as:

$$(h_j, w_j) = (h_j^0, w_j^0) + \Delta_s^j(k_c, k_c, :), \quad (13)$$

where (k_c, k_c) is the center position of $K \times K$ square.

Condensation Operation. Given the estimated correspondence pixel (h_j, w_j) and other offsets in Δ_s^j , we can obtain the surrounding pixels and combine them with (h_j, w_j) to form $K \times K$ pixel set. Their coordinates are stored in $\mathbf{n}_s^j \in \mathbb{R}^{K \times K \times 2}$. This process can be formulated as:

$$\begin{aligned} \mathbf{n}_s^j(x, y, :) &= (h_j, w_j) + \Delta_s^j(x, y, :), \quad x, y \in \{1, \dots, K\}; x, y \neq k_c, \\ \mathbf{n}_s^j(k_c, k_c, :) &= (h_j, w_j), \end{aligned} \quad (14)$$

where (k_c, k_c) is the center position of $K \times K$ square.

Finally, we can perform the condensation operation via the index selection on $\mathbf{a}_s^j$, formulated as:

$$\tilde{\mathbf{a}}_s^j = \mathbf{a}_s^j(\mathbf{n}_s^j), \quad (15)$$

where $\tilde{\mathbf{a}}_s^j$ is the condensed representation of $\mathbf{a}_s^j$, *i.e.* pixel associations of (h_i, w_i) to image I_j . Furthermore, we also illustrate the condensation for pixel associations of (h_i, w_i) to all images, *i.e.* $\mathbf{a}_s \in \mathbb{R}^{N \times H_s \times W_s \times L_s}$, in Figure 2. By applying such a condensation process to all pixel associations, we can obtain the final condensed hyperassociation $\tilde{\mathbf{A}}_s$.

3.3 Object-aware Cycle Consistency Loss

To achieve accurate correspondence estimations, applying effective supervisions on them is necessary. As explicit semantic correspondence annotations are not available, we can only rely on unsupervised losses by imposing correspondence-related constraints on estimated correspondences. Previous unsupervised approaches apply constraints to all pixels [29, 34, 44], including those on the background and extraneous objects that don't have mutual correspondences, hence harming the model effectiveness. To avoid this problem, we propose an object-aware constraint to only access losses on the co-salient pixels.

We propose an Object-aware Cycle Consistency (OCC) loss for the supervision of correspondence estimations in CoSOD. The cycle consistency can be explained as: if a co-salient pixel (h_i, w_i) of image I_i corresponds to the pixel (h_j, w_j) in image I_j , then the pixel (h_j, w_j) should also semantically correspond to pixel (h_i, w_i) .

Based on this cycle consistency constraint, we adopt image warping operations to conduct the OCC loss. Specifically, we first warp the image I_i^s (a resized I_i to align the scales in the s -th stage) as $I_{i \rightarrow j}^s$ using the $I_i^s \rightarrow I_j^s$ correspondence estimations. Next, we warp $I_{i \rightarrow j}^s$ backward to $I_{i \rightarrow j \rightarrow i}^s$ using $I_j^s \rightarrow I_i^s$ correspondence estimations. Finally, we can utilize the SSIM loss between I_i^s and $I_{i \rightarrow j \rightarrow i}^s$ to measure the cycle consistency for mutual correspondence pixels in I_i^s and I_j^s . Moreover, to ensure the constraints are only conducted on co-salient objects, we mask the images with their ground truth masks, formulated as:

$$\mathcal{L}_s^C = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N \mathcal{L}_{SSIM}(I_i^s \cdot G_i^s, I_{i \rightarrow j \rightarrow i}^s \cdot G_i^s), \quad (16)$$

Table 1: Ablation Study of our proposed modules. SAG, SAC, and FCC are ablation modules for PAG, CAC, and OCC, respectively.

ID	Modules						CoCA [43]			
	SAG	PAG	SAC	CAC	FCC	OCC	$S_m \uparrow$	$E_\xi \uparrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$
1							0.6936	0.7642	0.5729	0.1206
2	✓						0.7236	0.8029	0.6357	0.1106
3		✓					0.7308	0.8122	0.6459	0.1075
4		✓		✓			0.7304	0.8123	0.6500	0.1085
5		✓		✓			0.7473	0.8155	0.6591	0.0956
6		✓		✓	✓		0.7398	0.8138	0.6506	0.1013
7		✓		✓		✓	0.7570	0.8248	0.6751	0.0924

where G_i^s is the resized ground truth of image I_i . The total OCC loss $\mathcal{L}^C$ is the sum of three stages, formulated as: $\mathcal{L}^C = \mathcal{L}_3^C + \mathcal{L}_4^C + \mathcal{L}_5^C$. More details can be found in the supplementary materials.

4 Experiments

4.1 Evaluation Datasets and Metrics

We follow [8] to evaluate our model on three benchmark datasets, *i.e.* CoCA [43] (1295 images of 80 groups), CoSal2015 [41] (2015 images of 50 groups), and CoSOD3k [7] (3316 images of 160 groups). We adopt four widely-used metrics for the quantitative evaluation, *i.e.* Structure-measure (S_m) [5], Maximum enhanced-alignment measure (E_ξ) [6], Maximum F-measure (F_β) [1], and Mean Absolute Error ($\mathcal{M}$) [3].

4.2 Implementation Details

To construct the training data, we follow [47] to use different combinations of three commonly used training datasets, *i.e.* DUTS class [43] (8250 images of 291 groups), COCO-9k [17] (9213 images of 65 groups), and COCO-SEG [36] (200,000 images of 78 groups), for a fair comparison with other state-of-the-art (SOTA) works. We also implement the synthesis strategy for the DUTS class dataset following [42].

For training specifics, we employ the data augmentation strategy in [20] and use 256×256 as the input size for the network. We employ the Adam optimizer [14] with $\beta_1 = 0.9$ and $\beta_2 = 0.99$ to optimize the network. We train our CONDA model for 300 epochs, starting with an initial learning rate of $1e - 4$, which is divided by 10 at the 60,000th iteration. Our experiments are implemented based on PyTorch [28] on a single Tesla A40 GPU, with the batchsize set to $N = 6$. The hyperparameter K in CAC is set to 9.

4.3 Ablation Study

We conduct ablation studies on the most challenging CoCA [43] dataset. To construct the baseline, we use the FPN [16] (with VGG-16 as encoder) as the

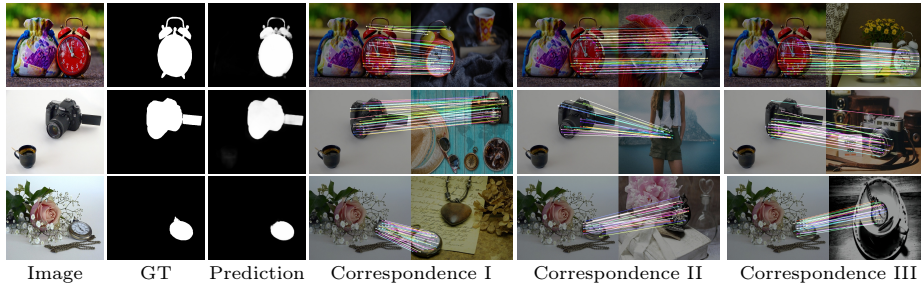


Fig. 4: Visual samples for the correspondence estimations. Correspondences I, II, and III visually display estimated correspondences between the main image and three related images. Sparse co-salient pixels were selected and connected to their corresponding pixels using colored lines for clear visualization.

foundational segmentation network and enhance it with the Region-to-Region correlation module (R2R) [15] to simply capture inter-image connections. Then, as shown in Table 1, we incrementally incorporate PAG, CAC, and OCC into the baseline for effectiveness analysis. We trained all ablation models with the DUTS class and COCO9k datasets.

Effectiveness of PAG. PAG first uses intermediate image features from the FPN encoder to calculate full-pixel hyperassociations. Then, aggregation networks are applied to generate deep association features for the decoder process. PAG effectively utilizes deep learning to model pair-wise pixel associations, achieving higher-level association knowledge compared to previous image feature optimization strategies. As shown in the 3rd line of Table 1, PAG shows significant performance boosts compared to the baseline model, with respective gains of 3.72%, 4.80%, 7.30%, and 1.31% in S_m , E_ξ , F_β , and $\mathcal{M}$.

Furthermore, to validate our approach of progressively enhancing image features with the previously generated association feature for improving hyperassociation calculation, we conduct an ablation experiment called Separate Association Generation (SAG) where association features are generated for three stages without image feature enhancements. As shown in the 2nd&3rd lines in Table 1, PAG outperforms SAG, indicating that our progressive enhancement design can obtain better hyperassociation.

Effectiveness of CAC. CAC aims to condense full-pixel hyperassociations in PAG by selecting corresponding pixels and their surrounding contexts. Results in the 3rd&5th lines of Table 1 show that introducing the CAC module improves the performance. Moreover, it reduces the multiply-accumulate operations (MACs) of aggregation networks from 91.38G in the full-pixel PAG to 77.19G¹. This indicates that utilizing correspondence estimation to condense the hyperassociations not only effectively reduces the computational burden but also helps obtain more accurate pixel associations.

We also analyze CAC in detail. As deep association features are necessary for reliable correspondence estimation, CAC first pre-condenses the hyperasso-

¹We input a group of 6 related 256x256 images to measure MACs.

Table 2: Quantitative comparison of our model with other SOTA methods. DC, C9, and CS are DUTS class, COCO9k, and COCO-SEG training data, respectively. **bold** and underline mark the best and second-best excellent results, respectively.

Methods	Training Set	CoCA [43]				CoSal2015 [41]				CoSOD3k [7]			
		$S_m \uparrow$	$E_\xi \uparrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$S_m \uparrow$	$E_\xi \uparrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$S_m \uparrow$	$E_\xi \uparrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$
GICD	DC	0.658	0.718	0.513	0.126	0.844	0.887	0.844	0.071	0.797	0.848	0.770	0.079
GCoNet	DC	0.673	0.760	0.544	<u>0.110</u>	0.845	0.888	0.847	0.068	0.802	0.860	0.778	0.071
GCoNet+	DC	<u>0.691</u>	0.786	<u>0.574</u>	0.113	<u>0.875</u>	<u>0.918</u>	<u>0.876</u>	<u>0.054</u>	<u>0.828</u>	0.881	<u>0.807</u>	<u>0.068</u>
CONDA	DC	0.717	<u>0.774</u>	0.600	0.102	0.890	0.926	0.894	0.049	0.832	<u>0.873</u>	0.807	0.067
ICNet	C9	0.654	0.704	0.513	0.147	<u>0.857</u>	0.901	<u>0.858</u>	0.058	0.794	0.845	0.762	0.089
DCFM	C9	0.710	0.783	0.598	0.085	0.838	0.893	<u>0.856</u>	0.067	0.809	<u>0.874</u>	<u>0.805</u>	0.067
GCoNet+	C9	<u>0.717</u>	<u>0.798</u>	<u>0.605</u>	0.098	0.853	0.902	0.857	0.073	<u>0.819</u>	0.877	0.796	0.075
CONDA	C9	0.730	0.801	0.622	<u>0.092</u>	0.865	0.910	0.875	<u>0.059</u>	0.825	0.877	0.810	<u>0.068</u>
CADC	DC+C9	0.680	0.744	0.549	0.133	0.867	0.906	0.865	0.064	0.815	0.854	0.778	0.088
DMT	DC+C9	0.725	0.800	0.619	0.108	<u>0.897</u>	<u>0.936</u>	<u>0.905</u>	<u>0.045</u>	<u>0.851</u>	<u>0.895</u>	<u>0.835</u>	<u>0.063</u>
GCoNet+	DC+C9	<u>0.734</u>	<u>0.808</u>	<u>0.626</u>	0.088	0.876	0.920	0.880	0.057	0.839	0.894	0.822	0.064
CONDA	DC+C9	0.757	0.825	0.675	<u>0.092</u>	0.904	0.940	0.912	0.042	0.857	0.899	0.844	0.060
UGEM	DC+CS	0.726	0.808	0.599	0.096	<u>0.885</u>	<u>0.935</u>	0.882	<u>0.051</u>	<u>0.853</u>	0.911	0.829	<u>0.060</u>
GCoNet+	DC+CS	<u>0.738</u>	<u>0.814</u>	<u>0.637</u>	0.081	0.881	0.926	<u>0.891</u>	0.055	0.843	0.901	<u>0.834</u>	0.061
CONDA	DC+CS	0.763	0.839	0.685	<u>0.089</u>	0.900	0.938	0.908	0.045	0.862	<u>0.911</u>	0.853	0.056

ciations using a maximum similarity approach to obtain initial deep association features and then performs further condensation based on the correspondences predicted by these initial deep association features. The hyperassociation condensation process with only the pre-condensation operation is called Similarity-induced Association Condensation (SAC). In Table 1, SAC only brings slight performance improvements due to the heuristic nature of the correspondence estimation. Nevertheless, SAC can provide the initial association feature for CAC to predict reliable correspondence.

Effectiveness of OCC. OCC provides self-supervision for CAC to enable more precise correspondence estimation. Results in the 5th&7th lines of Table 1 show OCC further improves the performance by leveraging more precise correspondence estimations to condense hyperassociations effectively. In addition, we conducted an ablation experiment to validate our object-aware design in CAC by replacing OCC with full-pixel cycle consistency (FCC) loss. Comparing the 5th&6th lines of Table 1, FCC leads to a notable performance decrease due to background pixels disrupting the correspondence learning.

Visualization of Correspondence Estimations. We present some visual samples of correspondence estimations for some co-salient pixels in Figure 4. Our semantic correspondence estimations are meaningful and can effectively depict the common attributes of co-salient objects at the pixel level.

4.4 Comparison with State-of-the-Art Methods

We compare our model with eight recent SOTA methods, *i.e.* GICD [43], ICNet [12], GCoNet [8], CADC [42], DCFM [40], DMT [15], UGEM [37], and GCoNet+ [47]. We directly utilize their officially released saliency maps for comparison. To ensure fairness, we trained our model with different combinations of three training datasets, following [47], to align with other compared methods. We

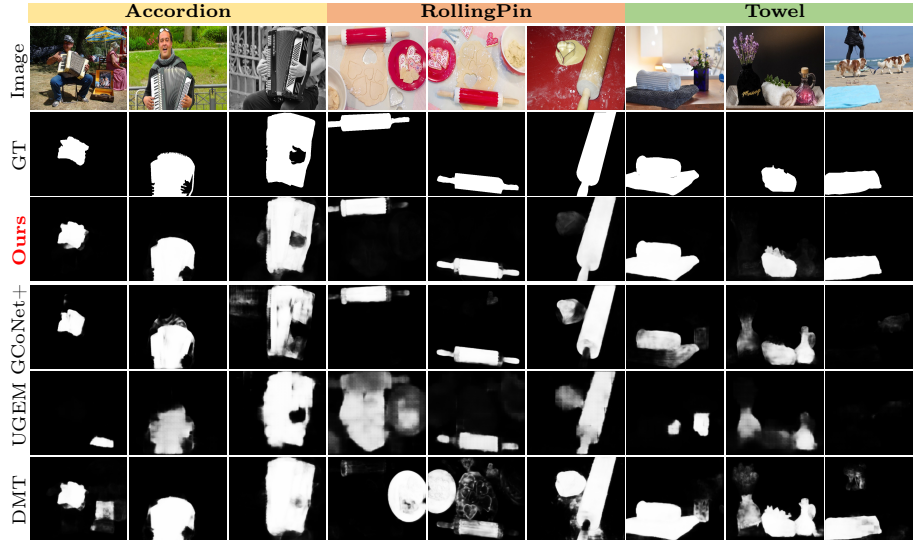


Fig. 5: Qualitative comparisons of our model with other SOTA methods.

denote three training datasets, *i.e.* DUTS class [43], COCO9k [17], and COCO-SEG [36], as DC, C9, and CS, respectively, for convenience. Our training sets include DC, C9, DC+C9, DC+CS. As shown in Table 2, we can observe that our model achieves the best performance with each training set in most benchmark datasets. What is even more exciting is that we achieve excellent results in the most challenging CoCA dataset, surpassing the second-best models by large margins, *e.g.* 2.5% S_m , 2.5% E_ξ , and 4.8% F_β with the DC+CS training set.

We also present visual comparisons in Figure 5. Our model can accurately detect co-salient objects in complex scenarios, such as irregularly shaped accordions accompanied by extraneous objects (people). However, other models easily fail to accurately segment co-salient objects.

5 Conclusion

This paper proposes a deep association learning strategy for CoSOD that directly embeds hyperassociations into deep association features. Correspondence estimation is also introduced to condense hyperassociations, enabling the exploration of pixel-level correspondences for CoSOD. We also utilize an object-aware cycle consistency loss to further refine correspondence estimations. Extensive experiments have verified the effectiveness of our method.

Acknowledgments: This work was supported in part by Key-Area Research and Development Program of Guangdong Province (No.2021B0101200001), the National Key R&D Program of China under Grant 2021B0101200001, the National Natural Science Foundation of China under Grant 62136007, 62036011, U20B2065, 6202781, 62036005, the Key R&D Program of Shaanxi Province under Grant 2021ZDLGY01-08, and the MBZUAI-WIS Joint Program for AI Research under Grants WIS P008 and P009.

References

1. Achanta, R., Hemami, S., Estrada, F., Susstrunk, S.: Frequency-tuned salient region detection. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 1597–1604 (2009)
2. Chen, W., Xu, H., Zhou, Z., Liu, Y., Sun, B., Kang, W., Xie, X.: Cost-former: Cost transformer for cost aggregation in multi-view stereo. *arXiv preprint arXiv:2305.10320* (2023)
3. Cheng, M.M., Warrell, J., Lin, W.Y., Zheng, S., Vineet, V., Crook, N.: Efficient salient region detection with soft image abstraction. In: *Int. Conf. Comput. Vis.* pp. 1529–1536 (2013)
4. Cho, S., Hong, S., Jeon, S., Lee, Y., Sohn, K., Kim, S.: Cats: Cost aggregation transformers for visual correspondence **34**, 9011–9023 (2021)
5. Fan, D.P., Cheng, M.M., Liu, Y., Li, T., Borji, A.: Structure-measure: A new way to evaluate foreground maps. In: *Int. Conf. Comput. Vis.* pp. 4548–4557 (2017)
6. Fan, D.P., Gong, C., Cao, Y., Ren, B., Cheng, M.M., Borji, A.: Enhanced-alignment measure for binary foreground map evaluation. In: *IJCAI*. pp. 698–704 (2018)
7. Fan, D.P., Lin, Z., Ji, G.P., Zhang, D., Fu, H., Cheng, M.M.: Taking a deeper look at co-salient object detection. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 2919–2929 (2020)
8. Fan, Q., Fan, D.P., Fu, H., Tang, C.K., Shao, L., Tai, Y.W.: Group collaborative learning for co-salient object detection. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 12288–12298 (2021)
9. Fu, H., Cao, X., Tu, Z.: Cluster-based co-saliency detection. *IEEE Trans. Image Process.* **22**(10), 3766–3778 (2013)
10. Hong, S., Cho, S., Nam, J., Lin, S., Kim, S.: Cost aggregation with 4d convolutional swin transformer for few-shot segmentation. In: *Eur. Conf. Comput. Vis.* pp. 108–126 (2022)
11. Hong, S., Nam, J., Cho, S., Hong, S., Jeon, S., Min, D., Kim, S.: Neural matching fields: Implicit representation of matching fields for visual correspondence **35**, 13512–13526 (2022)
12. Jin, W.D., Xu, J., Cheng, M.M., Zhang, Y., Guo, W.: Icnnet: Intra-saliency correlation network for co-saliency detection. *Adv. Neural Inform. Process. Syst.* (2020)
13. Kim, S., Min, J., Cho, M.: Transmatcher: Match-to-match attention for semantic correspondence. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 8697–8707 (2022)
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014)
15. Li, L., Han, J., Zhang, N., Liu, N., Khan, S., Cholakkal, H., Anwer, R.M., Khan, F.S.: Discriminative co-saliency and background mining transformer for co-salient object detection. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 7247–7256 (2023)
16. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 2117–2125 (2017)
17. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: *Eur. Conf. Comput. Vis.* pp. 740–755 (2014)
18. Liu, J.J., Hou, Q., Cheng, M.M., Feng, J., Jiang, J.: A simple pooling-based design for real-time salient object detection. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 3917–3926 (2019)

19. Liu, N., Han, J.: Dhsnet: Deep hierarchical saliency network for salient object detection. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 678–686 (2016)
20. Liu, N., Han, J., Yang, M.H.: Picanet: Learning pixel-wise contextual attention for saliency detection. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3089–3098 (2018)
21. Liu, N., Luo, Z., Zhang, N., Han, J.: Vst++: Efficient and stronger visual saliency transformer. IEEE Trans. Pattern Anal. Mach. Intell. (2024)
22. Liu, N., Zhang, N., Wan, K., Shao, L., Han, J.: Visual saliency transformer. In: Int. Conf. Comput. Vis. pp. 4722–4732 (2021)
23. Liu, Y., Zhu, L., Yamada, M., Yang, Y.: Semantic correspondence as an optimal transport problem. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 4463–4472 (2020)
24. Liu, Y., Liu, N., Yao, X., Han, J.: Intermediate prototype mining transformer for few-shot semantic segmentation **35**, 38020–38031 (2022)
25. Luo, Z., Liu, N., Zhao, W., Yang, X., Zhang, D., Fan, D.P., Khan, F., Han, J.: Vscore: General visual salient and camouflaged object detection with 2d prompt learning. In: IEEE Conf. Comput. Vis. Pattern Recog. (2024)
26. Min, J., Kang, D., Cho, M.: Hypercorrelation squeeze for few-shot segmentation. In: Int. Conf. Comput. Vis. pp. 6941–6952 (2021)
27. Moon, S., Sohn, S.S., Zhou, H., Yoon, S., Pavlovic, V., Khan, M.H., Kapadia, M.: Hm: Hybrid masking for few-shot segmentation. In: Eur. Conf. Comput. Vis. pp. 506–523. Springer (2022)
28. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library **32** (2019)
29. Shen, X., Darmon, F., Efros, A.A., Aubry, M.: Ransac-flow: generic two-stage image alignment. In: Eur. Conf. Comput. Vis. pp. 618–637. Springer (2020)
30. Shi, X., Huang, Z., Bian, W., Li, D., Zhang, M., Cheung, K.C., See, S., Qin, H., Dai, J., Li, H.: Videoflow: Exploiting temporal cues for multi-frame optical flow estimation. arXiv preprint arXiv:2303.08340 (2023)
31. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: Int. Conf. Learn. Represent. (2015)
32. Su, Y., Deng, J., Sun, R., Lin, G., Su, H., Wu, Q.: A unified transformer framework for group-based segmentation: Co-segmentation, co-saliency detection and video salient object detection. IEEE Trans. Multimedia (2023)
33. Sun, G., Liu, Y., Tang, H., Chhatkuli, A., Zhang, L., Van Gool, L.: Mining relations among cross-frame affinities for video semantic segmentation. In: Eur. Conf. Comput. Vis. pp. 522–539. Springer (2022)
34. Truong, P., Danelljan, M., Yu, F., Van Gool, L.: Warp consistency for unsupervised learning of dense correspondences. In: Int. Conf. Comput. Vis. pp. 10346–10356 (2021)
35. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Adv. Neural Inform. Process. Syst. **30** (2017)
36. Wang, C., Zha, Z.J., Liu, D., Xie, H.: Robust deep co-saliency detection with group semantic. In: AAAI. vol. 33, pp. 8917–8924 (2019)
37. Wu, Y., Song, H., Liu, B., Zhang, K., Liu, D.: Co-salient object detection with uncertainty-aware group exchange-masking. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 19639–19648 (2023)

38. Xiong, Z., Li, H., Zhu, X.X.: Doubly deformable aggregation of covariance matrices for few-shot segmentation. In: *Eur. Conf. Comput. Vis.* pp. 133–150. Springer (2022)
39. Xu, G., Wang, X., Ding, X., Yang, X.: Iterative geometry encoding volume for stereo matching. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 21919–21928 (2023)
40. Yu, S., Xiao, J., Zhang, B., Lim, E.G.: Democracy does matter: Comprehensive feature mining for co-salient object detection. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 979–988 (2022)
41. Zhang, D., Han, J., Li, C., Wang, J.: Co-saliency detection via looking deep and wide. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 2994–3002 (2015)
42. Zhang, N., Han, J., Liu, N., Shao, L.: Summarize and search: Learning consensus-aware dynamic convolution for co-saliency detection. In: *Int. Conf. Comput. Vis.* pp. 4167–4176 (2021)
43. Zhang, Z., Jin, W., Xu, J., Cheng, M.M.: Gradient-induced co-saliency detection. In: *Eur. Conf. Comput. Vis.* pp. 455–472 (2020)
44. Zhao, D., Song, Z., Ji, Z., Zhao, G., Ge, W., Yu, Y.: Multi-scale matching networks for semantic correspondence. In: *Int. Conf. Comput. Vis.* pp. 3354–3364 (2021)
45. Zhao, J.X., Liu, J.J., Fan, D.P., Cao, Y., Yang, J., Cheng, M.M.: Egnet: Edge guidance network for salient object detection. In: *Int. Conf. Comput. Vis.* pp. 8779–8788 (2019)
46. Zhao, W., Zhang, J., Li, L., Barnes, N., Liu, N., Han, J.: Weakly supervised video salient object detection. In: *IEEE Conf. Comput. Vis. Pattern Recog.* pp. 16826–16835 (2021)
47. Zheng, P., Fu, H., Fan, D.P., Fan, Q., Qin, J., Tai, Y.W., Tang, C.K., Van Gool, L.: Gconet+: A stronger group collaborative co-salient object detector. *IEEE Trans. Pattern Anal. Mach. Intell.* (2023)
48. Zhuge, M., Fan, D.P., Liu, N., Zhang, D., Xu, D., Shao, L.: Salient object detection via integrity learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**(3), 3738–3752 (2022)