Deep Unsupervised Part-Whole Relational Visual Saliency

Yi Liu[a], Xiaohui Dong[a], Dingwen Zhang[b,*] and Shoukun Xu[a,*]

[a]School of Computer Science and Artificial Intelligence, Aliyun School of Big Data, and School of Software, Changzhou University, Changzhou, Jiangsu 213000, China

[b]School of Automation, Northwestern Polytechnical University, Xian, Shaanxi 710129, China

ARTICLE INFO

Keywords:
Unsupervised salient object detection
Part-object relationship
Consistency-aware fusion strategy

ABSTRACT

Deep Supervised Salient Object Detection (SSOD) excessively relies on large-scale annotated pixel-level labels which consume intensive labour acquiring high quality labels. In such precondition, deep Unsupervised Salient Object Detection (USOD) draws public attention. Under the framework of the existing deep USOD methods, they mostly generate pseudo labels by fusing several hand-crafted detectors' results. On top of that, a Fully Convolutional Network (FCN) will be trained to detect salient regions separately. While the existing USOD methods have achieved some progress, there are still challenges for them towards satisfactory performance on the complex scene, including 1) poor object wholeness owing to neglecting the hierarchy of those salient regions; 2) unsatisfactory pseudo labels causing by unprimitive fusion of hand-crafted results. To address these issues, in this paper, we introduce the property of part-whole relations endowed by a Belief Capsule Network (BCNet) for deep USOD, which is achieved by a multi-stream capsule routing strategy with a belief score for each stream within the CapsNets architecture. To train BCNet well, we generate high-quality pseudo labels from multiple hand-crafted detectors by developing a consistency-aware fusion strategy. Concretely, a weeding out criterion is first defined to filter out unreliable training samples based on the inter-method consistency among four hand-crafted saliency maps. In the following, a dynamic fusion mechanism is designed to generate high-quality pseudo labels from the remaining samples for BCNet training. Experiments on five public datasets illustrate the superiority of the proposed method. Codes have been released on: https://github.com/Mirlongue/Deep-Unsupervised-Part-Whole-Relational-Visual-Saliency.

1. Introduction

Salient Object Detection (SOD) aims at identifying and segmenting the attractive regions in a given image. Due to its property, SOD has been widely used in a range of research aspects and applications, *e.g.*, segmentation [1, 2], image fusion [3], image retrieval [4], and object recognition [5]. In the deep learning era, deep Supervised SOD (SSOD) has achieved significant progress over the traditional hand-crafted methods [6]. However, existing deep SSOD methods excessively rely on the high-quality pixel-level annotation labels, which consume heavy labour and hardly cover all the natural scenes. Alternatively, deep Unsupervised SOD (USOD) can cast off the reliance on the annotation labels. In this paper, we focus on deep USOD.

There have been some attempts for deep USOD to this day. For example, Zhang *et al.* [7] make the earliest effort for USOD without using pixel-level human annotations, which generates the pseudo labels from hand-crafted detectors in intra-image and inter-method consistency perspectives and trains on DHSNet [8]. Afterwards, a few works [9, 10, 11] activate the development of deep USOD. These works mostly generate pseudo labels from hand-crafted methods through enforcing inter-image consistency and then train their models. During the training, multiple hand-crafted saliency results will be updated separately through using the deep network, which are used to generate updated pseudo labels for a new training epoch. To improve the quality of pseudo labels, the uncertainty is mined for each hand-crafted method for further refining pseudo labels. On top of pseudo labels, a Fully Convolutional Network (FCN) is trained to detect salient objects by learning discriminative saliency cues. While the existing USOD methods have achieved some progress, their performance is still far from being satisfactory. This bottleneck is derived from two folds: i) The simple FCN architecture that usually adopts a backbone without specific designs has limited power to capture the discriminative features, *e.g.*, FCN detecting the high-contrast salient regions may cause poor object wholeness, as shown in Fig. 1(a); ii) The non-satisfactory quality of pseudo labels degrades the network training.

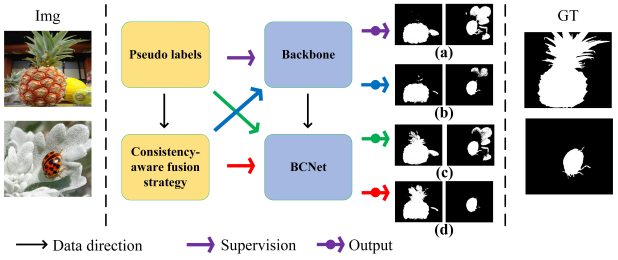

Figure 1: Motivation statement. The pseudo labels is adopted from [10] as the initialization. BCNet and consistency-aware dynamic fusion help to capture the object wholeness and filter background. Their integration achieves a further improvement.

To address the first problem, we introduce the property of part-whole relations endowed by CapsNets [12] into the task of deep USOD. Rather than a direct and primitive

*Equally corresponding authors:
zhangdingwen2006yyy@gmail.com (D. Zhang); jpuxsk@163.com (S. Xu)

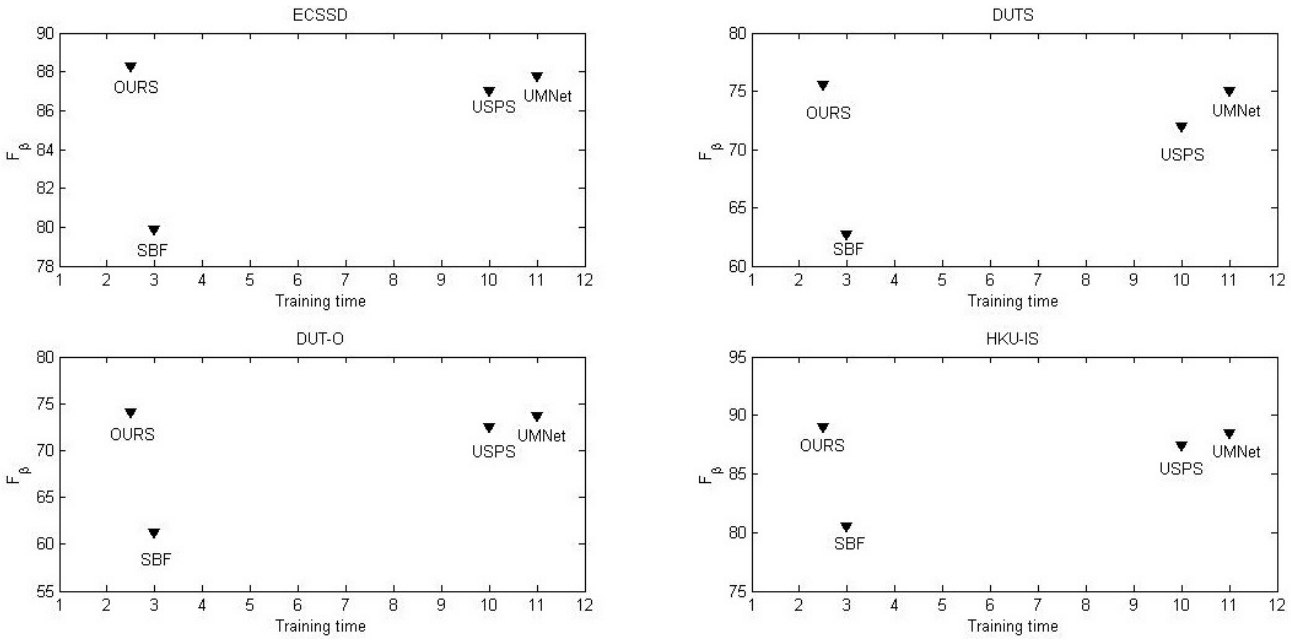

Figure 2: Performance and training time (hour) for different deep USOD methods. It can be seen that we achieve better performance with less training time, compared with the other deep USOD methods.

embedding, we design a Belief CapsNet (BCNet) to capture the object wholeness for salient objects. Concretely, a multi-stream sparse capsule routing strategy is designed to reduce the complexity of the capsule routing. A confidence score is additionally computed for each stream to highlight those important streams while suppressing those noisy streams. As shown in Fig. 1(c), our BCNet can improve the object wholeness by enhancing the inner details capture and background noise suppressing.

To address the second problem, *i.e.*, training our BCNet for deep USOD, we propose a consistency-aware fusion strategy to generate high-quality pseudo labels. First, the intersection and union operations are used to define two image-level consistency criteria, including weeding out criteria and fusion criteria. The former aims to weed out unreliable training samples, enhancing the robustness of the network training. The latter is used to select the fusion methods, including the union fusion strategy for high inter-method consistency and the dynamic fusion algorithm for low inter-method consistency[1]. Especially, under the circumstance of low inter-method consistency, an iterative dynamic fusion algorithm is designed to generate high-quality labels. In the iterative dynamic fusion algorithm, we define a label quality measure metric to calculate the similarity between saliency map of each hand-crafted method and the current pseudo label, which is activated by softmax to compute the balanced weight for each hand-crafted method for further updating the current pseudo label. The generated high-quality pseudo labels using our consistency-aware fusion strategy are used to train BCNet to generate better saliency inference. As shown in Fig. 1(b), our consistency-aware fusion induced pseudo labels reduce the background noise significantly. Further to say, the integration of our two designations makes the saliency predictions closer to the ground truth (see Fig. 1(d)) and achieves better performance with less training time (see Fig. 2), compared with the other deep USOD methods.

To sum up, the contributions of the paper can be concluded as:

(1) The property of part-whole relations is introduced in the task of deep USOD. To the best of our knowledge, this is the first attempt to employ part-whole relations for deep USOD.

(2) A BCNet architecture is designed to capture the object wholeness with a confidence multi-stream strategy, which reduces the implementation complexity and enhances the part-whole relations representation.

(3) A consistency-aware fusion strategy is proposed to generate high-quality pseudo labels, which help to enhance the performance of network training.

The remainder is organized as follows. In Sec. 2, we review the related works. Then, details of our method are disclosed in Sec. 3. Sec. 4 presents complete experimental results to demonstrate the advantages of our method and we draw conclusions in Sec. 5.

2. Related Work

In this section, we focus on those works related ours, including deep SSOD, deep weakly SSOD, deep USOD and CapsNets.

[1]The two fusion methods, union fusion strategy and dynamic fusion algorithm for high/low inter-method consistency are included in the module called dynamic fusion mechanism. Their relationship will be clearly discussed in the Sec. 3.3.2

2.1. Deep SSOD

In 2014, Han *et al.* [13] first introduced deep learning for salient object detection, which was achieved by a deep reconstruction network. Henceforth, deep learning, especially CNNs, sweeps across the field of SOD. For example, Li *et al.* [14] extracted multi-scale deep features for saliency detection. Wang *et al.* [15] proposed to learn local and global saliency cues to detect the salient object. Liu *et al.* [8] designed a two-stage network, which generated coarse saliency predictions and refinement, respectively. Zhang *et al.* [16] aggregated multi-level feature maps into multiple resolutions to detect the salient object. Wu *et al.* [17] designed a cross refinement unit to refine multi-level features simultaneously. Zhang *et al.* [18] proposed an attention network to selectively integrate multi-level context information for saliency prediction. Wang *et al.* [19] implemented salient object detection with the helps of pyramid attention and the task of salient edge detection. Zhao *et al.* [20] designed a context-aware pyramid feature extraction module for capturing rich saliency context features for further saliency prediction. Wang *et al.* [21] conducted top-down and bottom-up saliency inference in a joint and iterative manner inspired by human perceptual processes. Wang *et al.* [22] proposed an attentive saliency network with efficient recurrent mechanism to refine saliency features from fixation map. Wei *et al.* [23] decomposed the saliency map into two parts, including one focusing on objetcs' center areas and another concentrating on edge details. Pang *et al.* [24] interacted adjacent-layer features via mutual learning and self-interaction to capture the multi-scale information. Tu *et al.* [25] embedded the edge prior in a hierarchical manner to learn boundary-aware saliency maps. Hu *et al.* [26] recurrently translated and aggreated context using a spatial attenuation context module. Ke *et al.* [27] fused efficiently contour and saliency using a recursive strategy.

Deep SSOD relies on large amounts of pixel-level manual annotations, which consume intensive labour. Differently, our USOD can tackle the demand for large-scale manual annotations, which will help to enhance the generalization of SOD for various scenes.

2.2. Deep Weakly SSOD

While deep SSOD methods have achieved significant improvements, they rely on large-scale high-quality annotations, which consume huge labour cost. Deep Weakly SSOD provide a feasible solution, which uses low-cost labels to achieve a balance between performance and annotations cost. For example, Wang *et al.* [28] supervised the network with image-level labels. Li *et al.* [29] carried out the task of salient object detection using the weak contour knowledge. Zeng *et al.* [30] adopted diverse weak supervision sources to provide enough information in training, including image-level tags, image captions, and unlabelled data. Zhang *et al.* [31] used scribble annotations to relabel the salient object detection dataset for model training. Zhang *et al.* [32] attempted to train the salient object detection network with a few training images. Piao *et al.* [33] proposed a multi-filter directive network to extract and filter saliency cues from noisy pseudo labels.

Those existing deep weakly SSOD methods still require an amount of non-pixel-level manual annotations, such as image level labels. Differently, our USOD network can further discard the dependence on manually annotated labels.

2.3. Deep USOD

Further, deep USOD was proposed, which can implement SOD without human annotation labels. Zhang *et al.* [7] began this task via an intra-image and inter-image fusion to build the saliency detection network from hand-crafted approaches. In the following, several attempts have been devoted to deep USOD. For example, Zhang *et al.* [9] built a noise model to fit the gaussian distribution, which helped to generate better pseudo labels from several hand-crafted methods. Zhang *et al.* [34] exploited the model consistency to identify inliers and outliers in noisy labels to infer high-quality pseudo labels for further network training. Nguyen *et al.* [10] performed saliency detection via refinement from hand-crafted methods with semantic information pretrained on other vision tasks. Wang *et al.* [11] built an uncertainty mining network to parse pseudo labels.

Different from the previous deep USOD methods that train a simple FCN without additional designs, our work incorporates the property of part-whole relations in deep USOD and designs a BCNet to enhance the wholeness of salient objects. Besides, we weed out some negative samples for pseudo labels generation, which is neglected by the previous deep USOD detectors.

2.4. CapsNets

While CNNs have achieved promising performance for visual recognition by identifying the existence of object parts, they will be fooled by a simple spatial structure disturbance. To this end, CapsNets [35, 12] design clever dynamic routing algorithms to capture the part-whole relationships in an image to enhance the equivalence of the network. Many attempts have been devoted to CapsNets architectures [36, 37, 38]. For example, Lenssen *et al.* [36] defined a group equivariant capsule layer to enhance the property of equivariance to CapsNets. Rajasegaran *et al.* [38] built a deep architecture to CapsNets employing 3D convolution.

In light of the excellent property of CapsNets, they have been successfully embedded in the task of saliency detection. For instance, Liu *et al.* [39] employed CapsNets to visual saliency, in which CapsNets were utilized to explore the part-whole relationships in the image to achieve the whole object saliency. Later, they consolidated their work by further using the capsule maps as guidance to learn more primitive saliency cues [40]. Zhuge *et al.* [41] utilized CapsNets for a part-whole verification module to enhance the agreement between parts and objects. Zhang *et al.* [42] engaged the complementary between contrast cues learned by CNNs and part-whole relations discovered by CapsNets to detect the salient object. Liu *et al.* [43] embedded CapsNets in a two-stage encoder-decoder structure for the task of camouflaged object detection, which is a SOD related task.

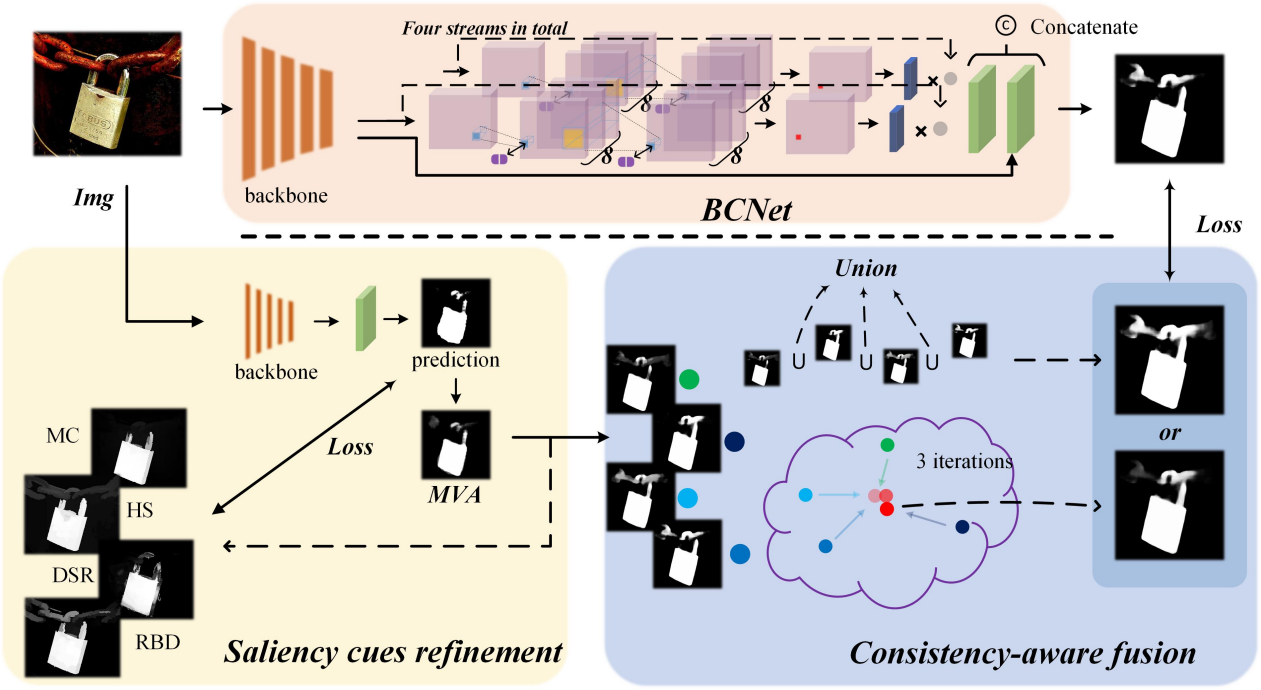

Figure 3: Overview of the proposed USOD framework. The input is fed into a backbone to learn deep backbone features, which are further input in a Belief CapsNet (BCNet) to capture the object wholeness semantics by exploring part-whole relations. In parallel, the deep backbone features are fed into the saliency cues refinement module to refine the hand-crafted saliency maps. The following consistency-aware fusion strategy generates high-quality pseudo labels to train the network.

Different from the existing CapsNets based SOD methods that focus on deep SSOD, our work tackles the problem of deep USOD with an efficient embedding of CapsNets, in which a confident multi-stream architecture is proposed to highlight those primitive capsule routing streams induced part-whole relations.

3. Proposed method

In this section, we elaborate details about the proposed framework. As shown in Fig. 3, the entire framework consists of three components, including saliency cues refinement, Belief CapsNet (BCNet), and consistency-aware fusion for pseudo labels generation. Specifically, the input is fed into a backbone network, *e.g.*, ResNet-101 [44], to learn deep backbone features, which are further input in BCNet to capture the object wholeness semantics by exploring part-whole relations. To get pseudo labels for training BCNet, the input image and four hand-crafted methods' saliency maps are fed into the refinement module, which is utilized to refine four hand-crafted methods (MC [45], HS [46], DSR [47], RBD [48]), and the consistency-aware fusion module, which is developed to generate high-quality labels via exploring the consistency among several hand-crafted methods.

3.1. Saliency cues refinement

For a fair comparison of our model with the previous methods, in accordance with the previous methods, *e.g.*, SBF [7], USPS [10], and UMNet [11], we select four traditional hand-crafted methods, including MC [45], HS [46], DSR [47], and RDB [48][2], to generate the coarse saliency maps.

The selected saliency maps are refined separately in the saliency cues refinement module. Specifically, we choose ResNet-101 [44] as our backbone network to learn deep backbone features, which are further fed into the Historical Moving Average (MVA) module [10]. This can be formulated as

$$MVA^{i} = (1-\lambda) \times MVA^{i-1} + \lambda \times p^{i-1}, \tag{1}$$

where p^{i-1} is the saliency prediction of epoch $(i-1)$. MVA^{i-1} is the forward pass result at epoch $(i-1)$ and the MVA^{0} is 0. λ is a balance parameter. Similar to [10], we repeat Eq. (1) for 25 epochs to obtain the refined saliency maps (MVA^{25}).

In Historical Moving Average (MVA) module of [10], they add fully connected Conditional Random Field (CRF) on the saliency prediction p of epochs. Different to them, we discard CRF, which achieves 27 hours reduced.

3.2. BCNet for salient object detection

Details of BCNet are described in Fig. 4, which consist of three components, including multi-stream strategy, stream confidence score and part-whole relational attention.

[2]The selected methods are definitely consistent with those of the previous unsupervised methods [7, 10, 11].

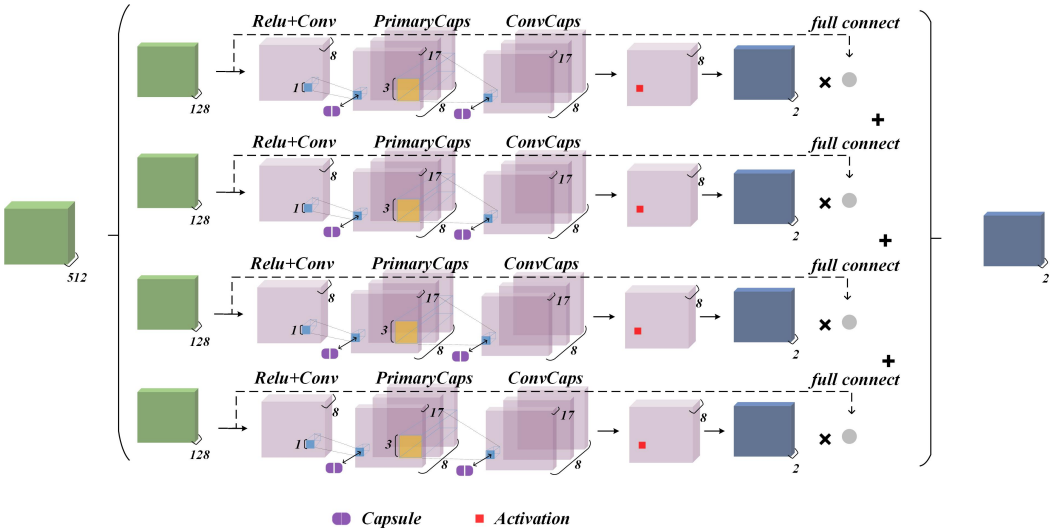

Figure 4: Details of BCNet. A multi-stream strategy is designed for CapsNets, which is further improved with a confidence score in each stream to highlight those important streams while suppressing those confusing streams.

3.2.1. Multi-stream strategy

The input feature maps ($44 \times 44 \times 512$) are divided into four streams along the channel dimension, which are fed into CapsNets [12] to learn the part-whole relations within each stream separately. Specifically, each stream contains one (ReLu + Conv (with stride of 2)) layer, one Primary Capsule (*PrimaryCaps*) layer, and one Convolutional Capsule (*ConvCaps*) layer[3]. In each stream, the last *ConvCaps* layer outputs two types of capsules ($22 \times 22 \times 2 \times 17$) corresponding to salient capsule and background capsule, respectively. This ensures a sparse routing for reducing the complexity, compared with the original capsule network [12].

3.2.2. Stream confidence score

Different streams sharing the same architecture learn different part-whole relational knowledge due to their different input features, which will cause different contributions of different streams to the final saliency detection. To highlight those important streams while suppressing those confusing streams, we compute a confidence score for each stream to measure its contribution. To achieve this, the input feature maps ($44 \times 44 \times 128$) are used to compute a score value via a convolution with stride of the feature maps scale, *i.e.*, 44. The confident score will be multiplied with the output capsule activation values of *ConvCaps* to obtain the confident part-whole relational semantics.

3.2.3. Part-whole relational attention

Multi-stream confident part-whole relational semantics are upsampled and integrated via addition to achieve the part-whole saliency prediction Sal^{PO} ($44 \times 44 \times 2$), which can be used to guide the backbone features to learn better saliency cues. First, the backbone feature maps Sal^{BB} ($44 \times 44 \times 2$) can be learned by a convolution on the backbone feature maps. On top of that, the part-whole relational attention can be written as

$$Sal^{Att} = f_{co}\left(f_{cc}\left(Sal^{BB}, Sal^{PO}\right), 2\right), \tag{2}$$

where $f_{co}(\cdot, 2)$ and $f_{cc}(\cdot)$ represent the operations of convolution with output channel of 2 and concatenation, respectively. Sal^{Att} is upsampled to the original input resolution as the final saliency map.

3.2.4. Difference to TSPOANet [39]

The main difference between our BCNet and TSPOANet [39] lies in three folds: i) Our BCNet incorporates the part-whole relations for deep USOD while TSPOANet [39] focusing on deep SSOD; ii) Our BCNet designs a multi-stream strategy to further reduce the complexity of capsule routing over the two-stream strategy of TSPOANet [39]; iii) A confidence score for each stream helps to highlight those important streams while suppressing unimportant streams, which is ignored by TSPOANet [39] that performs a fully-connected capsule routing to integrate two streams with high complexity.

3.3. Consistency-aware fusion

The refined hand-crafted saliency maps in Eq. (1) will be integrated to generate high-quality pseudo labels. To achieve this, a consistency-ware fusion strategy is proposed, which consists of two steps, including weeding out negative samples and dynamic fusion mechanism. Details will be described in the following.

3.3.1. Weeding out negative samples

Different hand-crafted methods possess different saliency priors, which will cause the inconsistency between four hand-crated methods. As shown in Fig. 5, low inter-method consistency implies an unreliable label, which will confuse

[3] These layers can be referred to TSPOANet [39].

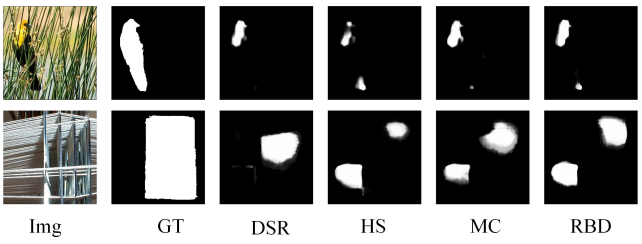

Figure 5: Negative samples with low inter-method consistency.

the network training and degrade the performance. To address this problem, we design a consistency-aware weeding out mechanism to filter out the negative samples. First, we define the inter-method consistency involving the detected salient regions for four hand-crafted detectors, *i.e.*, salient regions detected by only one method (f_1), only two methods (f_2), only three methods (f_3), and only four methods (f_4), respectively. The consistency can be written as

$$\begin{aligned} f_4 &= \cap\left(\left\{Sal_i, \forall i\right\}, 4\right), \\ f_3 &= \cup\left\{\cap\left(\left\{Sal_i, \forall i\right\}, 3\right)\right\} - f_4, \\ f_2 &= \cup\left\{\cap\left(\left\{Sal_i, \forall i\right\}, 2\right)\right\} - f_3 - f_4, \\ f_1 &= \cup\left(\left\{Sal_i, \forall i\right\}, 4\right) - f_2 - f_3 - f_4, \end{aligned} \tag{3}$$

where Sal_i is the saliency map of the hand-crafted method i. $\cup$ and $\cap$ represent the union and intersection operations, respectively. $\cup\left(\left\{Sal_i, \forall i\right\}, n\right)$ and $\cap\left(\left\{Sal_i, \forall i\right\}, n\right)$ represent the union and intersection of any n methods, respectively.

The weeding out criterion is formulated as

$$\sum_{i=3}^{4} f_i \Big/ \sum_{i=1}^{4} f_i < \mu, \tag{4}$$

where μ is the inconsistency threshold. Eq. (4) will weed out those low inter-method consistency samples. In this paper, μ is set to 0.4, which weeds out 1% samples of the entire training samples.

To give a comprehensive understanding of f_1, f_2, f_3, and f_4, Fig. 6 shows their visualization results of two examples. As shown from the top image in Fig. 6, hand-crafted methods infer in-consistent saliency maps, which cannot well identify the salient object. Differently, from the bottom image of Fig. 6, hand-crafted methods infer consistent saliency maps, which segment the salient object well. Besides, as shown in Fig. 6, f_1 detects very few salient regions, such as only the object boundary in the bottom image, while f_4 focusing on the regions identified by four methods simultaneously. Based on the weeding out criterion, the top image is weeded out from the training set due to $\sum_{i=3}^{4} f_i \Big/ \sum_{i=1}^{4} f_i < 0.4$, while the bottom image being saved in the training set due to $\sum_{i=3}^{4} f_i \Big/ \sum_{i=1}^{4} f_i > 0.4$.

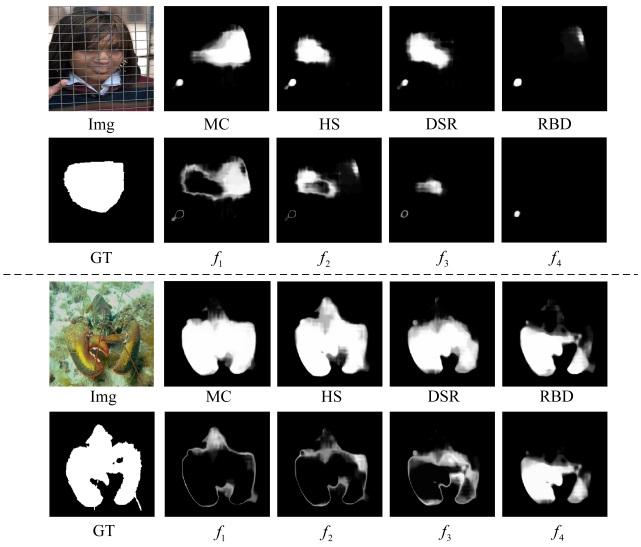

Figure 6: Visualizations for the weeding out criterion. Top and bottom images are negative and positive samples, respectively.

3.3.2. Dynamic fusion mechanism

From the remaining hand-crafted saliency maps, we design a dynamic fusion mechanism to generate high-quality pseudo labels. To this end, a fusion criterion based on the inter-method consistency is first defined. On top of the fusion criterion, we design two fusion mechanisms, including the union fusion strategy and the dynamic fusion algorithm[4], to generate pseudo labels.

Component 1: Fusion criterion. As the definitation of Eq. (3), f_1 and f_3 are defined as the inter-method consistency from one method and three methods, respectively. It is obvious that f_1 and f_3 define a low inter-method consistency and a high inter-method consistency, respectively. More concretely, f_1 means those salient regions that are detected by only one of foure methods, while f_3 refers to those salient regions that are detected by three of four methods. Therefore, the salient regions of f_1 will be less confident than the salient regions of f_3. Inspired by this observation, the salient regions of f_1 and f_3 can be treated as the low-confidence and high-confidence salient regions, respectively. On top of that, we define the fusion criterion as $\frac{f_1}{f_3}$, which reveals the inter-method consistency between four hand-crafted saliency maps. Specifically, a small/high $\frac{f_1}{f_3}$ means a high/low inter-method consistency. For the low $\frac{f_1}{f_3}$, we choose the simple fusion, *i.e.*, union fusion, to generate pseudo labels. For the high $\frac{f_1}{f_3}$, we design a dynamic fusion algorithm to generate pseudo labels. A threshold θ is defined to measure the inter-method consistency degree. In this paper, we define θ as $\frac{5}{8}$.

[4]Dynamic fusion mechanism is higher-level than dynamic fusion algorithm in terms of concept.

Algorithm 1 Dynamic fusion algorithm. fus is the fused result, *i.e.*, the pseudo label, of each iteration. $w_i (i = 1, 2, 3, 4)$ are the balanced weights for different hand-crafted methods, including MC [45], HS [46], DSR [47], RBD [48]. $softmax$ is the activation function.

Procedure Dynamic fusion algorithm (Sal_i)
Initialization:
$w_1 = w_2 = w_3 = w_4 = 1/4$,
for t iterations **do**
1. $fusion\_step$:
$$fus = \sum_{i=1}^{4} \left(Sal_i \times w_i\right),$$
2. $weight\_step$:
$$d_i = dis\left(Sal_i, fus\right),$$
$$w_i = softmax\left(d_i\right).$$
end

Component 2: Union fusion strategy. If $\frac{f_1}{f_3} < \theta$, a high inter-method consistency is assumed for four hand-crafted saliency maps. In this case, we adopt the union operation of four hand-crafted saliency maps to generate the pseudo labels, *i.e.*, f_4.

Component 3: Dynamic fusion algorithm. If $\frac{f_1}{f_3} \geq \theta$, a low inter-method consistency is assumed for four hand-crafted saliency maps. In this case, a dynamic fusion algorithm is proposed to integrate four hand-crafted saliency maps to generate high-quality labels.

A label quality measure metric is first defined as

$$dis\left(x, y\right) = -\ln\left(1 - F_\beta(x, y)^2\right). \tag{5}$$

$F_\beta(x, y)$ is defined as

$$F_\beta\left(x, y\right) = \left(1 + \beta^2\right) \frac{x \times y}{\beta^2 \times x + y}, \tag{6}$$

where $\beta^2 = 3$.

$F_\beta(x, y)$ in Eq. (6) measures the similarity between x and y. Similarly, the similarity between saliency map of hand-crafted method i and the current pseudo label fus can be formulated as

$$d_i = dis\left(Sal_i, fus\right). \tag{7}$$

The balanced weight for Sal_i when fusing can be written as

$$w_i = softmax\left(d_i\right). \tag{8}$$

Algorithm 1 details the dynamic fusion algorithm. At the initialization, the average weight is used to set for the balanced weight w_i. On top of that, two steps, including $fusion\_step$ and $weight\_step$, are iterated to generate the pseudo label via fusion and compute the balanced weights, respectively. Several iterations can achieve labels with high quality. In this paper, 3 iterations have achieved not-bad pseudo labels, which can be visually found in Fig. 7.

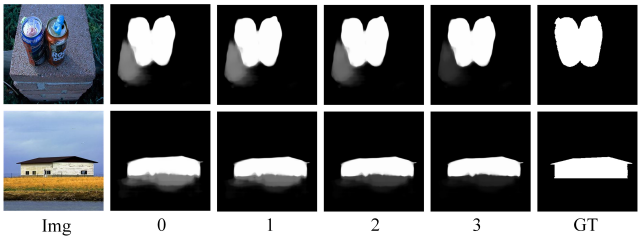

Figure 7: Pseudo labels of each iteration for Algorithm 1. With the increasing iterations, the noise is degraded for a pure saliency map.

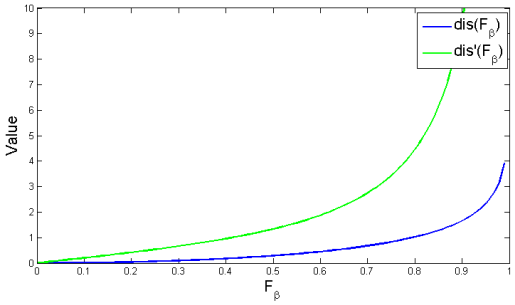

Figure 8: Curves of the function and its derivative of Eq. (5). It can be seen that the derivative beyond 0.85 is large, which helps to discriminate the consistency between four hand-crafted saliency maps that have F_β being around 0.85.

Insight into the label quality measure metric. The curves of Eq. (5) and its derivative are described in Fig. 8. The dis' is the derivative of the $dis(x, y)$ function in Eq. (5). We draw their curves to illustrate dist has a significant derivative when $F_\beta > 0.85$. That is, when $F_\beta > 0.85$, our $dis(x, y)$ function in Eq. (5) can tackle their subtle changes of F_β. Fortunately, the selected hand-crafted methods, including MC [45], HS [46], DSR [47], and RBD [48], share the values of F_β beyond 0.85. Therefore, our $dis(x, y)$ function in Eq. (5) can tackle the integration of these hand-crafted methods.

Difference to the existing pseudo labels generations. The main difference between our pseudo labels generation strategy and that of the existing deep USOD methods lies in: i) We weed out some negative training samples, which is neglected by the existing methods and may degrade their network training; ii) We propose a dynamic fusion algorithm to substitute for the complex CRF to compute the pseudo labels on top of MVA, while the most existing methods, *e.g.*, USPS [10] and UMNet [11], utilizing CRF on top of MVA to generate the pseudo labels.

3.4. Loss function

Similar to USPS [10], we adopt the image-level loss function w.r.t each training example to supervise BCNet for training. The image-level loss function can be written as

$$L = 1 - F_\beta\left(p, pl\right), \tag{9}$$

Table 1
Ablation Analysis. The best and second best methods are marked by **bold** and underline for each part, respectively. "CAF" means consistency-aware fusion.

Component Settings	ECSSD [46]				DUTS [28]				DUT-O [49]				HKU-IS [14]			
	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$
(a) Performance (%) of different components.																
Base	6.19	87.57	86.40	90.26	6.98	73.90	79.07	84.18	6.60	72.85	79.54	84.09	4.07	88.18	88.08	92.77
+ BCNet	**5.97**	87.94	86.85	90.45	6.76	74.61	79.55	84.39	6.36	73.46	79.99	84.31	3.91	88.58	88.47	92.99
+ CAF	6.27	87.92	86.30	89.90	6.58	75.46	79.49	84.30	**6.18**	**74.17**	79.94	84.18	4.03	88.89	88.18	92.67
+ BCNet + CAF	6.01	**88.26**	**86.91**	**90.71**	**6.56**	**75.51**	**80.10**	**85.24**	6.30	74.02	**80.23**	**84.57**	**3.87**	**88.97**	**88.62**	**93.28**
(b) Performance (%) comparison: w/o CRF *vs* w/ CRF.																
w/ CRF	6.36	87.55	85.73	89.36	6.73	74.20	78.38	82.87	6.19	73.06	79.03	82.63	4.16	88.41	87.42	91.98
w/o CRF	**6.01**	**88.26**	**86.91**	**90.71**	**6.56**	**75.51**	**80.10**	**85.24**	**6.30**	**74.02**	**80.23**	**84.57**	**3.87**	**88.97**	**88.62**	**93.28**
(c) Performance (%) of different fusion strategies.																
Union	**5.74**	87.67	**87.28**	**91.42**	7.14	73.03	79.49	84.53	6.95	71.66	79.42	83.75	4.00	87.33	88.34	93.24
Dynamic fusion algorithm	6.34	87.92	86.16	89.77	6.62	75.22	79.37	84.07	**6.16**	**74.28**	80.01	84.21	4.05	88.90	88.12	92.56
Dynamic fusion mechanism	6.01	**88.26**	86.91	90.71	**6.56**	**75.51**	**80.10**	**85.24**	6.30	74.02	**80.23**	**84.57**	**3.87**	**88.97**	**88.62**	**93.28**
(d) Performance (%) of different versions of CapsNets.																
Original CapsNet [12]	**5.99**	88.08	86.83	90.45	**6.53**	**75.58**	79.90	84.73	**6.18**	**74.07**	80.22	84.33	**3.85**	88.95	88.54	93.03
TSPOANet [39]	6.04	88.02	86.77	90.47	6.67	75.03	79.66	84.54	6.32	73.72	79.94	84.24	3.93	88.74	88.45	93.05
BCNet	6.01	**88.26**	**86.91**	**90.71**	6.56	75.51	**80.10**	**85.24**	6.30	74.02	**80.23**	**84.57**	3.87	**88.97**	**88.62**	**93.28**
(e) Performance (%) of our pseudo labels on different baselines.																
TSPOANet-Pseudo	7.33	85.14	85.99	89.45	8.15	67.32	78.95	81.00	8.53	65.20	77.98	79.50	5.03	84.38	87.41	91.73
CPD-Pseudo	7.61	84.38	86.19	88.23	8.79	68.68	78.47	81.24	8.89	66.91	77.77	79.99	5.65	83.76	87.13	90.33
BCNet	**6.01**	**88.26**	**86.91**	**90.71**	**6.56**	**75.51**	**80.10**	**85.24**	**6.30**	**74.02**	**80.23**	**84.57**	**3.87**	**88.97**	**88.62**	**93.28**
(f) Performance (%) comparison: w/o confidence score *vs.* w/ confidence score in BCNet.																
w/o confidence score	**5.96**	88.13	86.88	90.45	6.60	75.23	79.78	84.48	**6.23**	73.95	80.15	84.53	3.88	88.83	88.52	92.98
BCNet	6.01	**88.26**	**86.91**	**90.71**	**6.56**	**75.51**	**80.10**	**85.24**	6.30	**74.02**	**80.23**	**84.57**	**3.87**	**88.97**	**88.62**	**93.28**
(g) Ablation study for the quality of our pseudo labels. "USPS/UMNet-Pseudo" means using our pseudo labels to train USPS/UMNet.																
USPS [10]	**6.11**	87.00	85.66	88.75	6.57	71.98	77.17	80.11	**5.70**	72.51	79.03	81.17	4.21	87.45	86.68	90.64
UMNet [11]	6.36	87.74	86.77	89.89	6.67	74.99	80.27	84.48	6.31	73.67	80.47	83.92	4.12	88.41	88.65	92.67
USPS/UMNet-Pseudo	6.15	**87.84**	**86.83**	**90.01**	**6.56**	**75.36**	**80.47**	**84.55**	6.18	**74.02**	**80.50**	**84.28**	**3.89**	**88.86**	**88.74**	**92.99**

where p and pl represent the forward prediction and the pseudo label obtained. F_β is defined in Eq. (6). In Eq. (9) and $\beta^2 = 2$. L is a linear loss and more robust to outliers and noise compared to high-order losses such as mean square error.

4. Experiment and Analysis

In this section, we will conduct experiment to understand and verify the proposed method.

4.1. Implementation details

We implement our experiments on two GTX 3090 Ti GPUs. We use the adam optimizer [50] with the momentum of 0.9, learning rate of 1e-6 and batch size of 20. The training set of MSRA-B [51] is chosen as the training dataset. MC [45], HS [52], DSR [47], RBD [48] are selected as four handcrafted methods. The image is resized to 352×352.

4.2. Dataset and evaluation metric

We evaluate the performance of our model on five benchmark datasets, details of which are described as follows.

4.2.1. Dataset

ECSSD [46] contains 1000 images collected from the Internet. These images are with complicated structures. **DUT-O** [49] has 5168 images with different sizes and complex structures. The backgrounds are very complicated to stand out the salient objects. **HKU-IS** [14] consists of 4447 images with multiple disconnected objects. It is divided into 3000 training images and 1447 test images. We evaluate our methods and other state-of-the-art methods on the test datasets. **DUTS** [28] contains 10533 training images and 5019 test images. The images in this dataset are with different scenes and various sizes. We use the test dataset to evaluate our model and the compared methods. **PASCAL-S** [53] includes 850 images describing various scenes.

4.2.2. Evaluation criteria

We evaluate the performance of our model as well as other state-of-the-art methods from both visual and quantitative perspectives. The quantitative metrics include F-measure, Mean Absolute Error (MAE), S-measure, and E-measure. Given a continuous saliency map, a binary mask B is achieved by thresholding. Precision is defined as $Precision = |B \cap G|/|B|$, and recall is defined as $Recall = |B \cap G|/|G|$, where G is the corresponding ground truth.

F-measure is an overall performance indicator, which is computed by

$$F_\beta = \frac{\left(1+\beta^2\right) Precision \times Recall}{\beta^2 Precision + Recall}. \tag{10}$$

As suggested in [54], $\beta^2 = 0.3$.

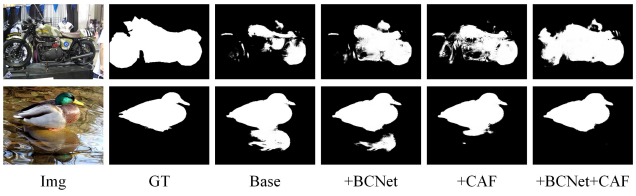

Figure 9: Visual comparison of each component of the proposed network. BCNet and CAF can both suppress background noise and get better object wholeness. The entire network (BCNet + CAF) enhances the performance with respect to each individual component.

MAE is defined as

$$MAE = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} |S(i,j) - G(i,j)|, \qquad (11)$$

where W and H are the width and height of the image, respectively.

S-measure (S_m) [55] is computed by

$$S_m = \alpha S_o + (1-\alpha) S^r, \qquad (12)$$

where S_o and S_r represent the object-aware and region-aware structure similarities between the prediction and the ground truth, respectively. α is set to 0.5 [55].

E-measure (E_m) [56] combines local pixel values with the image-level mean value to jointly evaluate the similarity between the prediction and the ground truth.

4.3. Ablation study

In this subsection, we will conduct several ablation studies to analyze the proposed method.

4.3.1. Different components

To analyze the contributions of each component of the proposed network, Table 1(a) lists the performance of different components. In Table 1(a), "Base" is composed by the saliency cues refinement module as well as the backbone network and there are two findings: i) BCNet and CAF both improve the performance compared with the base model; ii) The entire network (BCNet + CAF) enhances the performance with respect to each individual component. These can be confirmed by Fig. 9.

4.3.2. w/o CRF vs *w/ CRF*

In our saliency cues refinement, we remove CRF from that of USPS [10], which is necessary for the previous methods. To verify the removal of CRF, Table 1(b) lists the performance using labels generated w/ and w/o CRF on BCNet. It can be seen that our pseudo labels without CRF beat those with CRF significantly on various datasets. Besides, due to removing CRF, our work saves 27 hours for saliency cues refinement, compared to USPS [10]. Fig. 10

Figure 10: Visual comparison of w/ CRF *vs.* w/o CRF. Left four columns, reveal that w/o CRF suppresses the background noise compared with w/ CRF. Besides, right two columns show that w/o CRF sharps the edge details of salient objects.

Figure 11: Generated labels of w/ CRF *vs.* w/o CRF. With the involvement of CRF, large background regions with high contrast will be judged as salient regions and object boundaries with high similarity to the background will be mistaken as background.

depicted visual comparisons of w/o CRF *vs.* w/CRF. Specifically, w/o CRF enhances the background noise suppression (left four columns of Fig. 10). Besides, w/o CRF sharps the edge details of saliency objects (right two columns of Fig. 10). The explanation for the superiority of w/o CRF can be seen in Fig. 11. As shown in Fig. 11, the involvement of CRF will make two problems: i) Large background regions with high contrast will be judged as salient regions (left four columns of Fig. 11); ii) Object boundaries with high similarity to the background will be mistaken as background (right four columns of Fig. 11). Due to these issues, w/o CRF generates better labels and infers better saliency maps in Fig. 10, compared with w/ CRF. Besides, Table 1(b) also tells that our pseudo labels beat the pseudo labels of USPS [10] (UMNet [11]) significantly[5], which is achieved by Table 1(b) "w/CRF".

[5] USPS [10] and UMNet [11] share the same pseudo labels. SBF [7] did not release their code, which is not involved here.

Table 2
Comparison of different μ (the number K of images wiped out). The best performance is marked by **bold**.

$\mu(K)$	ECSSD [46]				DUTS [28]				DUT-O [49]				HKU-IS [14]			
	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$
0(0)	6.02	88.05	86.71	90.53	6.64	75.16	79.74	84.80	6.38	73.57	79.84	84.20	3.94	88.74	88.39	93.01
0.2(6)	6.08	88.17	86.75	90.50	6.69	75.02	79.70	84.83	6.35	73.81	80.01	84.48	3.92	88.88	88.45	93.14
0.4(20)	**6.01**	**88.26**	**86.91**	**90.71**	**6.56**	**75.51**	**80.10**	**85.24**	6.30	**74.02**	**80.23**	**84.57**	**3.87**	**88.97**	**88.62**	**93.28**
0.5(40)	6.17	88.09	86.56	90.47	6.67	74.91	79.83	84.74	6.33	73.68	79.85	84.37	4.01	88.65	88.35	92.93
0.6(91)	6.33	87.92	86.47	90.30	6.64	75.02	79.69	84.63	6.38	73.32	79.78	83.98	3.97	88.54	88.34	92.95
0.65(146)	6.43	87.78	86.18	89.83	6.57	75.11	79.57	84.37	**6.22**	73.83	79.93	84.12	4.05	88.33	88.07	92.63
0.7(128)	6.53	87.53	86.56	90.43	6.61	75.08	79.79	84.85	6.32	73.55	79.92	84.26	3.93	88.58	88.43	93.11

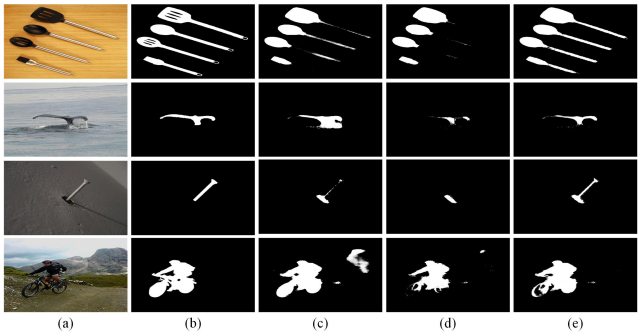

Figure 12: (a) Images; (b) GT; (c) Union fusion strategy; (d) Dynamic fusion algorithm; (e) Dynamic fusion mechanism. Visual comparison of different fusion strategies. Our dynamic fusion mechanism successfully coordinates the union fusion strategy and dynamic fusion algorithm and gets good saliency predictions.

4.3.3. Weeding out mechanism

To understand the contribution of the weeding out mechanism, we train our model with different μ under the same setting. Table 2 lists the performance of different μ on four datasets. Specifically, as shown in Table 2, the performance is improved when μ is increasing from 0 to 0.4, and the performance is decreased when μ is increasing from 0.4. Therefore, it is obvious that out model performs best when μ is 0.4. Inspired by this, μ is set to 0.4 in this paper.

4.3.4. Dynamic fusion mechanism

To verify the superiority of the proposed fusion strategy, we compare three versions of fusion, including only union fusion strategy, only dynamic fusion algorithm, and dynamic fusion mechanism. Table 1(c) lists the performance of different fusion strategies. It can be obviously seen that our consistency-aware achieves the best performance with respect to most of evaluation metrics. Besides, visual comparisons in Fig. 12 tell that the dynamic fusion strategy surpasses the union fusion and dynamic fusion algorithm separately with better wholeness and sharp object boundaries.

4.3.5. Iterations in dynamic fusion algorithm

To take a deep insight into the dynamic fusion algorithm, Table 3 lists the performance of different iterations. It can be seen that 3 iterations achieve the best performance with respect to various metrics. Inspired by this, 3 iterations is used in this paper.

Table 3
Dynamic fusion performance (%) of different iterations on ECSSD [46]. The best performance is marked by **bold**.

Iteration	0	1	2	3	4	5
$MAE\downarrow$	6.66	6.54	6.42	**6.34**	6.45	6.38
$F_\beta\uparrow$	87.53	87.66	87.85	**87.92**	87.88	87.84

4.3.6. BCNet

To understand the contribution of our BCNet, we compare several versions, including original CapsNet [12], two-stream strategy in TSPOANet [39], and BCNet, under the same settings. Table 1(d) lists the performance of different versions. It can be seen that BCNet beats the others on most of datasets. Besides, to probe into the effectiveness of our BCNet for deep USOD, we use our pseudo labels to train two existing deep SSOD networks, including TSPOANet [39] and CPD [57], named TSPOANet-Pseudo and CPD-Pseudo. As shown in Table 1(e), with our pseudo labels, the advantage of our BCNet over TSPOANet-Pseudo and CPD-Pseudo demonstrates the superiority of the architecture of BCNet for deep USOD.

Fig. 13 displays the visual results of BCNet, original CapsNets [12], and two-stream strategy in TSPOANet [39]. It demonstrates that BCNet gets better object wholeness and background suppressing, compared to the original CapsNet and TSPOANet. Besides, Fig. 14 shows visual examples using the pseudo labels on BCNet, the original CapsNet, and TSPOANet. Under the same pseudo labels, BCNet detects saliency maps more close to GT, compared to original CapsNet and TSPOANet.

To discover the importance of the confidence score, we take an experiment for BCNet with and without the confidence score. Which is list in Table 1(f). It can be found that BCNet with the confidence score can enhance the performance, which proves the effectiveness of the confidence score in BCNet. This benefits from that the confidence score helps to focus more on those informative streams of BCNet when inferring.

Table 4
Comparison of the number of streams (1-8) with the fixed total capsule types. The best performance is marked by **bold**.

Number of streams	ECSSD [46]				DUTS [28]				DUT-O [49]				HKU-IS [14]			
	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$
1	6.20	87.69	86.39	90.06	6.57	75.33	79.76	84.84	6.31	73.77	79.91	84.35	3.95	88.77	88.38	93.01
2	6.11	87.76	86.59	90.37	6.70	74.85	79.61	84.62	6.46	73.45	79.77	84.12	3.94	88.63	88.42	93.04
3	6.13	87.93	86.72	90.63	6.67	74.86	79.54	84.47	6.39	73.62	79.80	84.18	3.95	88.70	88.39	93.07
4	**6.01**	**88.26**	**86.91**	90.71	**6.56**	**75.51**	80.10	**85.24**	6.30	**74.02**	**80.23**	**84.57**	**3.87**	**88.97**	**88.62**	**93.28**
5	6.05	88.13	86.89	**90.75**	6.60	75.43	79.80	84.93	6.35	73.73	79.97	84.33	3.94	88.83	88.54	93.17
8	6.04	88.23	86.93	90.68	**6.56**	75.36	**80.12**	85.17	**6.28**	73.95	80.21	84.41	3.91	88.80	88.43	92.99

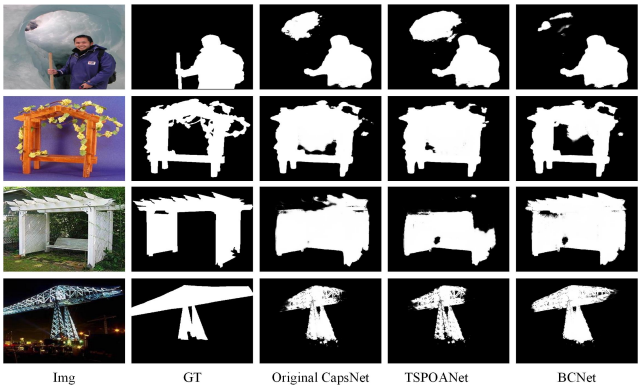

Figure 13: Visual comparison of different versions of CapsNets under the same settings. Our BCNet can better distinguish between background noise and salient objects as well as predicting complete salient objects.

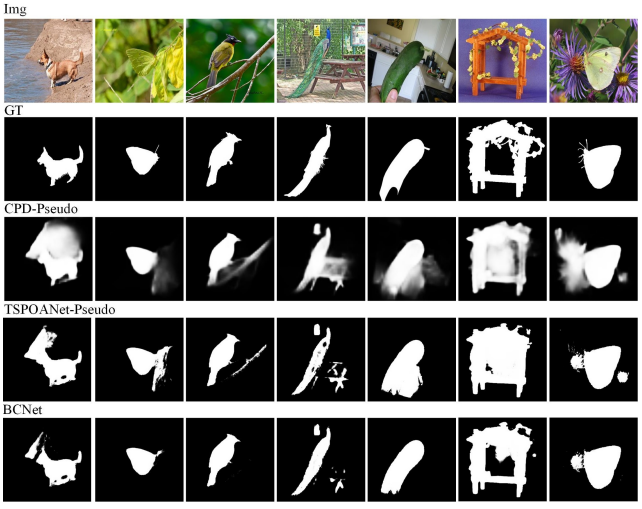

Figure 14: Visual comparison of different baselines using our pseudo labels. BCNet can better cater to those pseudo labels for deep USOD, compared with CPD and TSPOANet.

4.3.7. Number of streams

Table 4 lists the performance of different datasets in terms of different numbers of streams (from 1 stream to 8 streams) with the fixed total capsule types. It can be seen that too few or too many streams will drop the performance, which is because of: i) unformed part-whole relations using few capsules in each stream under the circumstance of too many streams; ii) noisy part-whole relations using many capsules in each stream under the circumstance of too few streams. From Table 4, the optimum number of streams is 4, which is used in this paper.

Table 5
The performance of pseudo labels under different values of θ in the dynamic fusion mechanism. The best performance is marked by **bold**.

θ	0	1/8	2/8	3/8	4/8	5/8	6/8	7/8	8/8
$MAE\downarrow$	4.63	4.63	4.61	4.57	4.53	**4.49**	**4.49**	**4.49**	**4.49**
$F_\beta\uparrow$	88.87	88.91	88.85	88.87	88.90	**88.92**	88.78	88.73	88.54

Table 6
Training time (hour) comparison. The most effient method is marked by **bold**.

	USPS [10]	UMNet [11]	Ours
Saliency cues refinement	41	41	**14**
Model training	10	11	**2.5**

4.3.8. Parameters

Under the setting of Table 1(d), the parameters of the original CapsNet [12], TSPOANet [39], and BCNet for the capsule routing are 71.62M, 40.64M, and 18.09M, respectively. It can be seen that our BCNet reduces the routing parameters significantly, which is owing to the multi-stream strategy.

4.3.9. The fusion criterion (θ) in the dynamic fusion mechanism

To discover the optimal value of θ in the dynamic fusion mechanism, we have conducted an ablation experiment for different values of θ to find the optimal value for performance. Table 5 lists the performance of different values of θ. It can be seen that when $\theta = 5/8$, our model performs best. Therefore, we select θ as 5/8 in this paper.

4.3.10. The quality of our pseudo labels

To better prove the quality of our pseudo labels over those of the previous unsupervised methods, we use our pseudo labels to train the previous unsupervised models, including USPS [10] and UMNet [11]. It is noted that USPS [10] and UMNet [11] share the same deep model. Table 1(g) lists the performance comparison of USPS [10] and UMNet [11] using different pseudo labels. It can be seen from Table 1(g), the performance using our pseudo labels to train the model is superior over that of USPS [10] and UMNet [11]

Table 7
Performance (%) of different methods on five benchmarks. The best performance for each group is marked by **bold**. - represents the authors did not provide the saliency map.

Method	ECSSD [46]				DUTS [28]				DUT-O [49]				PASCAL-S [53]				HKU-IS [14]			
	$MAE\downarrow$	$F_\beta\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_m\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_m\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_m\uparrow$	$S_m\uparrow$	$E_m\uparrow$	$MAE\downarrow$	$F_m\uparrow$	$S_m\uparrow$	$E_m\uparrow$
(a) Fully Supervised																				
PiCANet [58]	4.64	88.67	91.38	92.33	5.41	78.20	86.07	87.24	6.79	72.24	82.64	83.28	7.83	80.02	84.77	86.86	4.15	87.08	90.54	92.26
CPD [57]	4.02	91.15	91.02	93.77	4.29	82.44	86.66	90.20	5.67	73.85	81.77	84.50	7.21	82.30	84.46	88.25	3.32	89.58	90.45	94.24
BASNet [59]	3.70	91.68	91.62	**94.32**	4.76	82.24	86.56	89.54	**5.65**	**76.68**	**83.62**	**86.50**	7.58	81.77	83.80	87.86	3.29	90.36	90.77	94.30
ITSD [60]	4.01	91.01	91.42	93.75	4.23	83.23	**87.71**	90.56	6.32	75.24	82.88	85.28	6.81	**83.05**	**85.63**	89.15	3.46	89.40	90.68	93.95
MINet [24]	**3.62**	**91.87**	**91.91**	**94.32**	**3.94**	**83.49**	87.49	**90.67**	5.69	74.04	82.18	84.58	**6.39**	83.03	85.45	**89.36**	**3.03**	**90.55**	**91.39**	**94.65**
TSPOANet [39]	5.15	88.73	86.84	90.20	4.82	79.91	82.02	87.48	6.38	70.30	76.92	82.32	7.49	81.23	81.42	85.08	4.01	87.95	86.56	92.63
(b) Weakly Supervised																				
C2S [29]	5.93	**86.55**	**88.17**	**91.19**	6.64	73.58	**81.75**	85.46	7.90	67.43	77.98	81.34	12.81	**80.65**	**78.32**	**81.80**	–	–	–	–
MWS [30]	9.64	76.18	82.75	79.10	9.12	64.78	75.88	74.30	10.87	59.70	75.58	72.85	13.30	66.78	76.75	73.52	8.37	73.51	81.79	78.78
WSS [31]	**5.90**	**86.55**	86.55	91.11	**6.22**	**74.67**	80.34	**86.50**	**6.84**	**70.15**	**78.48**	**83.45**	13.99	78.84	74.95	79.75	**4.70**	**85.76**	**86.49**	**92.32**
MFNet [33]	9.20	82.03	82.29	85.44	8.86	69.02	76.81	80.57	11.37	58.73	71.33	75.42	**11.91**	73.22	76.72	80.22	6.70	81.85	83.49	87.39
(c) Handcrafted Unsupervised																				
DSR [47]	17.15	**58.19**	68.51	65.28	**14.78**	**47.85**	**65.21**	**64.04**	**13.88**	**50.62**	67.28	66.48	26.91	44.35	54.16	51.04	**14.22**	**58.84**	69.93	67.32
MC [45]	20.24	53.75	**69.24**	61.10	19.88	42.57	62.46	57.59	18.63	46.78	64.91	60.95	27.15	49.55	61.28	54.13	18.40	52.41	68.38	60.79
RBD [48]	**17.14**	56.15	68.84	**65.39**	15.31	45.63	64.66	63.60	14.38	50.04	**68.15**	**66.50**	**24.70**	53.08	**61.55**	**58.13**	14.24	57.20	**70.62**	**67.40**
HS [46]	22.75	56.73	68.51	62.03	24.32	42.70	60.06	57.60	22.74	47.26	63.26	60.48	28.73	**53.15**	61.38	55.20	21.50	54.62	67.42	62.17
(d) Deep Unsupervised																				
SBF [7]	8.80	79.84	83.23	85.01	10.69	62.70	68.61	71.54	10.76	61.20	74.73	76.32	13.09	69.51	75.79	77.77	7.53	80.50	82.91	89.33
USPS [10]	6.11	87.00	85.66	88.75	6.57	71.98	77.17	80.11	**5.70**	72.51	79.03	81.17	10.54	74.47	76.54	79.47	4.21	87.45	86.68	90.64
UMNet [11]	6.36	87.74	86.77	89.89	6.67	74.99	**80.27**	84.48	6.31	73.67	**80.47**	83.92	–	–	–	–	4.12	88.41	**88.65**	92.67
Ours	**6.01**	**88.26**	**86.91**	**90.71**	**6.56**	**75.51**	80.10	**85.24**	6.30	**74.02**	80.23	**84.57**	**10.38**	**77.18**	**78.35**	**82.69**	**3.87**	**88.97**	88.62	**93.28**

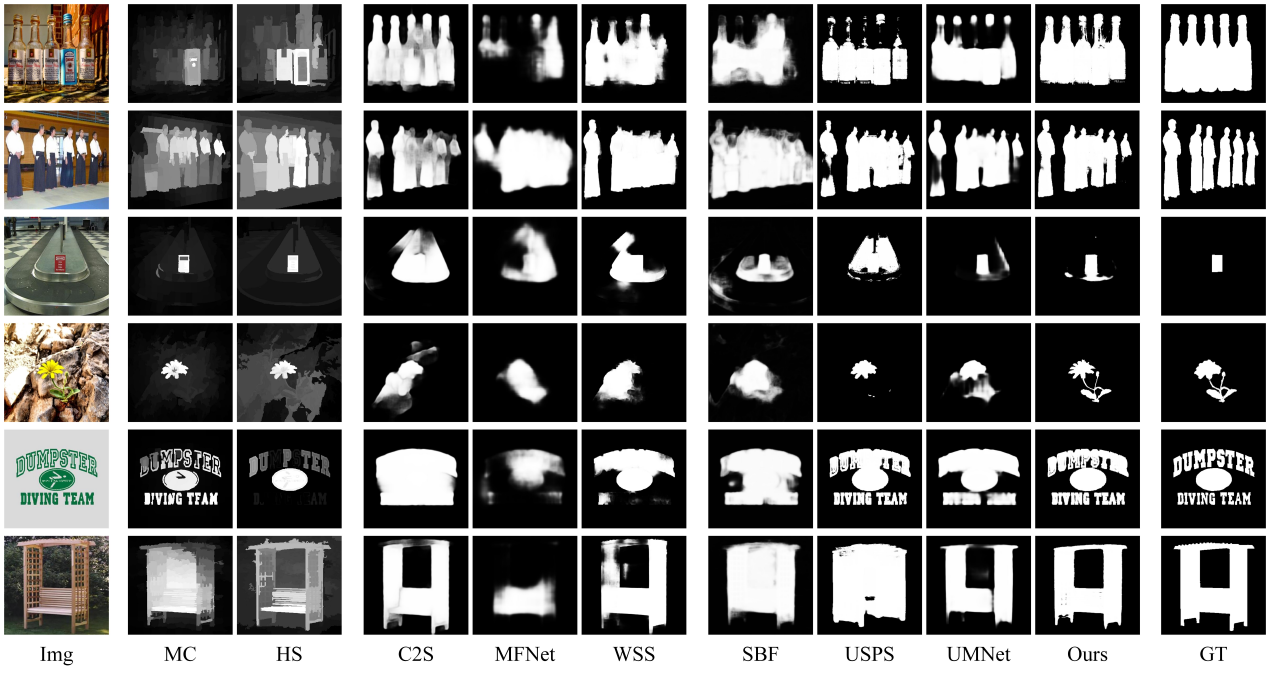

Figure 15: visual example of the saliency detection results obtained by our approach and other state-of-the-art methods. MC [45], HS [46] are hand-crafted methods. C2S [29], MFNet [33], and WSS [31] are weakly-SSOD methods. SBF [7], USPS [10], and UMNet [11] are USOD methods.

on four datasets, which proves the superior quality of our pseudo labels.

4.3.11. Training time comparision

Due to the fact that our model shares the similar training procedure with USPS [10] and UMNet [11], including saliency cues refinement and model training using the results of the saliency cues refinement, Table 6 lists the training time of USPS [10], UMNet [11], and our model. Specifically in Table 6, for the saliency cues refinement, due to the removal of CRF, we achieve 27 hours reduced compared to USPS [10] and UMNet [11]. Besides, using the results of the saliency cues refinement, our model training is efficient when comparing with USPS [10] and UMNet [11]. In summary, we perform efficiently with respect to the training time compared with the previous deep USOD methods.

4.4. Comparison with state-of-the-art methods

4.4.1. Quantitative comparison

To demonstrate the superiority of our deep USOD method, we select 17 SOD methods for comparison, including 3 state-of-the-art deep USOD methods (SBF [7], USPS [10], and UMNet [11]), 6 deep SSOD methods (PiCANet [58], CPD [57], BASNet [59], ITSD [60], MINet [24], TSPOANet [39]), 4 weakly SOD methods (C2S [29], MWS [30], WSS [31], and MFNet [33]), and 4 hand-crafted SOD methods (DSR [47], MC [45], RBD [48], HS [46]). Table 7 lists the performance of various methods on five benchmarks. As shown in Table 7, it can be seen that our method outperforms the state-of-the-art deep USOD methods (Table 7(d)) and hand-crafted methods (Table 7(c)) on most of evaluation metrics, which verifies the superiority of our method for deep USOD. Besides, our method beats some of weakly SOD methods (Table 7(b)), which further demonstrates the power of our deep method for salient object detection under unsupervison. When comparing the deep SSOD method, TSPOANet [39], and OURS, we find that our work is competitive and even better than TSPOANet [39] w.r.t F_β, S_m, and E_m on ECSSD [46], DUT-O [49], and HKU-IS [14], which demonstrates that our detector achieves competitive performance with the deep SSOD methods.

4.4.2. Visual comparison

Visual comparisons of different methods are shown in Fig. 15. For diverse scenes, including multiple objects (top two rows of Fig. 15), small object (middle two rows of Fig. 15), and large objects (bottom two rows of Fig. 15), our work obtains better object wholeness and accuracy, compared to the other methods. This benefits from the proposed high-quality pseudo labels and the powerful BCNet.

4.5. Failure cases

Fig. 16 displays some failure cases for our model. From the left two columns of Fig. 16, our model endowed by CapsNets can segment the objects, but cannot parse the image accurately. From the right two columns of Fig. 16, our model cannot segment the accurate object wholeness under the circumstance of complicated foreground and background. In the future, some cutting-edge techniques, *e.g.*, region-object relations [61] and spatial granularity [62], will be adopted to improve the robustness and feasibility of our method to various scenes.

5. Conclusion

In this paper, we have introduced the property of part-whole relations in the task of deep USOD with the designation of BCNet. Part-whole relations come from the routing process of CapsNets. However, the complex routing process of the previous CapsNets limits its widespread application for the large-scale dense prediction of salient objects. To enhance the ability of CapsNets for the segmentations of salient objects, in this paper, we have embeded two components in CapsNets. First, a multi-stream strategy with few capsules in each stream was utilized to ensure a sparse routing, which helped to reduce the complexity and noisy capsule assignments. Secondly, a confidence score has been computed for each stream to highlight those important streams while suppressing those confusing streams, which ensured using those primitive streams of capsule features to infer saliency. Besides, a consistency-aware dynamic fusion mechanism has been proposed to generate pseudo labels from hand-crafted salient object detectors. Various experiments have verified the superiority, and efficient ablation experiments have been studied to understand the proposed method. In the future, we will improve our work by involving other deep learning techniques, *e.g.*, hierarchical architecture [63], contrastive learning [64], transformer [65], SAM [66], SegGPT [67], diffusion [68], region-object relations [61] and spatial granularity [62], *etc*. Besides, we will attempt to apply our method to some real applications, such as visual localization [69], medical image segmentation [70], video object detection [71], remote scene [72] and urban scene [73], *etc*.

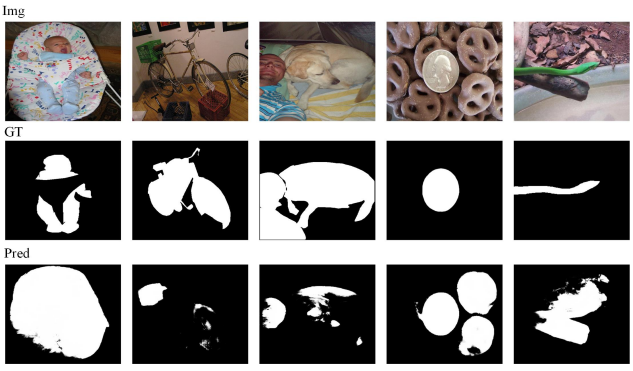

Figure 16: Failure cases.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant 62001341 and in part by the Natural Science Foundation of Jiangsu Province under Grant BK20221379.

References

[1] J. Han, K. N. Ngan, M. Li, H.-J. Zhang, Unsupervised extraction of visual attention objects in color images, IEEE Transactions on Circuits and Systems for Video Technology 16 (1) (2006) 141–145.

[2] S. J. Oh, R. Benenson, A. Khoreva, Z. Akata, M. Fritz, B. Schiele, Exploiting saliency for object segmentation from image level labels, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5038–5047.

[3] J. Han, E. J. Pauwels, P. De Zeeuw, Fast saliency-aware multi-modality image fusion, Neurocomputing 111 (2013) 70–80.

[4] P. Wang, J. Wang, G. Zeng, J. Feng, H. Zha, S. Li, Salient object detection for searched web images via global saliency, in: Proceedings of IEEE Confernece on Computer Vision and Pattern Recognition, 2012, pp. 3194–3201.

[5] D. Gao, S. Han, N. Vasconcelos, Discriminant saliency, the detection of suspicious coincidences, and applications to visual recognition,

IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (6) (2009) 989–1005.

[6] W. Wang, Q. Lai, H. Fu, J. Shen, H. Ling, R. Yang, Salient object detection in the deep learning era: An in-depth survey, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (6) (2021) 3239–3259.

[7] D. Zhang, J. Han, Y. Zhang, Supervision by fusion: Towards unsupervised learning of deep salient object detector, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 4048–4056.

[8] N. Liu, J. Han, Dhsnet: Deep hierarchical saliency network for salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 678–686.

[9] J. Zhang, T. Zhang, Y. Dai, M. Harandi, R. Hartley, Deep unsupervised saliency detection: A multiple noisy labeling perspective, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 9029–9038.

[10] T. Nguyen, M. Dax, C. K. Mummadi, N. Ngo, T. H. P. Nguyen, Z. Lou, T. Brox, Deepusps: Deep robust unsupervised saliency prediction via self-supervision, Advances in Neural Information Processing Systems 32 (2019).

[11] Y. Wang, W. Zhang, L. Wang, T. Liu, H. Lu, Multi-source uncertainty mining for deep unsupervised saliency detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11727–11736.

[12] G. E. Hinton, S. Sabour, N. Frosst, Matrix capsules with em routing, in: Proceedings of the International Conference on Learning Representations, 2018, pp. 3856–3866.

[13] J. Han, D. Zhang, X. Hu, L. Guo, J. Ren, F. Wu, Background prior-based salient object detection via deep reconstruction residual, IEEE Transactions on Circuits and Systems for Video Technology 25 (8) (2014) 1309–1321.

[14] G. Li, Y. Yu, Visual saliency based on multiscale deep features, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5455–5463.

[15] L. Wang, H. Lu, X. Ruan, M.-H. Yang, Deep networks for saliency detection via local estimation and global search, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3183–3192.

[16] P. Zhang, D. Wang, H. Lu, H. Wang, X. Ruan, Amulet: Aggregating multi-level convolutional features for salient object detection, in: Proceedings of the IEEE Conference on Computer Vision, 2017, pp. 202–211.

[17] Z. Wu, L. Su, Q. Huang, Stacked cross refinement network for edge-aware salient object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 7264–7273.

[18] X. Zhang, T. Wang, J. Qi, H. Lu, G. Wang, Progressive attention guided recurrent network for salient object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 714–722.

[19] W. Wang, S. Zhao, J. Shen, S. C. Hoi, A. Borji, Salient object detection with pyramid attention and salient edges, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 1448–1457.

[20] T. Zhao, X. Wu, Pyramid feature attention network for saliency detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 3085–3094.

[21] W. Wang, J. Shen, M.-M. Cheng, L. Shao, An iterative and cooperative top-down and bottom-up inference network for salient object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[22] W. Wang, J. Shen, X. Dong, A. Borji, R. Yang, Inferring salient objects from human fixations, IEEE Transactions on Pattern Analysis and Machine Intelligence 42 (8) (2020) 1913–1927. doi:10.1109/TPAMI.2019.2905607.

[23] J. Wei, S. Wang, Z. Wu, C. Su, Q. Huang, Q. Tian, Label decoupling framework for salient object detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 13025–13034.

[24] Y. Pang, X. Zhao, L. Zhang, H. Lu, Multi-scale interactive network for salient object detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 9413–9422.

[25] Z. Tu, Y. Ma, C. Li, J. Tang, B. Luo, Edge-guided non-local fully convolutional network for salient object detection, IEEE transactions on circuits and systems for video technology 31 (2) (2020) 582–593.

[26] X. Hu, C.-W. Fu, L. Zhu, T. Wang, P.-A. Heng, Sac-net: Spatial attenuation context for salient object detection, IEEE Transactions on Circuits and Systems for Video Technology 31 (3) (2020) 1079–1090.

[27] Y. Y. Ke, T. Tsubono, Recursive contour-saliency blending network for accurate salient object detection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 2940–2950.

[28] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, X. Ruan, Learning to detect salient objects with image-level supervision, in: Proccedings of the IEEE Conference Computer Vision and Pattern Recognition, 2017, pp. 136–145.

[29] X. Li, F. Yang, H. Cheng, W. Liu, D. Shen, Contour knowledge transfer for salient object detection, in: Proceedings of the european conference on computer vision (ECCV), 2018, pp. 355–370.

[30] Y. Zeng, Y. Zhuge, H. Lu, L. Zhang, M. Qian, Y. Yu, Multi-source weak supervision for saliency detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 6074–6083.

[31] J. Zhang, X. Yu, A. Li, P. Song, B. Liu, Y. Dai, Weakly-supervised salient object detection via scribble annotations, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 12546–12555.

[32] D. Zhang, H. Tian, J. Han, Few-cost salient object detection with adversarial-paced learning, Advances in Neural Information Processing Systems 33 (2020) 12236–12247.

[33] Y. Piao, J. Wang, M. Zhang, H. Lu, Mfnet: Multi-filter directive network for weakly supervised salient object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 4136–4145.

[34] J. Zhang, Y. Dai, T. Zhang, M. Harandi, N. Barnes, R. Hartley, Learning saliency from single noisy labelling: A robust model fitting perspective, IEEE Transactions on Pattern Analysis and Machine Intelligence 43 (8) (2020) 2866–2873.

[35] S. Sabour, N. Frosst, G. E. Hinton, Dynamic routing between capsules, in: Advances in Neural Information Processing Systems, 2017, pp. 3856–3866.

[36] J. E. Lenssen, M. Fey, P. Libuschewski, Group equivariant capsule networks, in: Advances in Neural Information Processing Systems, 2018, pp. 1–10.

[37] J. Chen, H. Yu, C. Qian, D. Z. Chen, J. Wu, A receptor skeleton for capsule neural networks, in: International Conference on Machine Learning, PMLR, 2021, pp. 1781–1790.

[38] J. Rajasegaran, V. Jayasundara, S. Jayasekara, H. Jayasekara, S. Seneviratne, R. Rodrigo, Deepcaps: Going deeper with capsule networks, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 10725–10733.

[39] Y. Liu, Q. Zhang, D. Zhang, J. Han, Employing deep part-object relationships for salient object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 1232–1241.

[40] Y. Liu, D. Zhang, Q. Zhang, J. Han, Part-object relational visual saliency, IEEE transactions on pattern analysis and machine intelligence 44 (7) (2021) 3688–3704.

[41] M. Zhuge, D.-P. Fan, N. Liu, D. Zhang, D. Xu, L. Shao, Salient object detection via integrity learning, IEEE Transactions on Pattern Analysis and Machine Intelligence 45 (3) (2022) 3738–3752.

[42] Q. Zhang, M. Duanmu, Y. Luo, Y. Liu, J. Han, Engaging part-whole hierarchies and contrast cues for salient object detection, IEEE Transactions on Circuits and Systems for Video Technology 32 (6) (2021) 3644–3658.

[43] Y. Liu, D. Zhang, Q. Zhang, J. Han, Integrating part-object relationship and contrast for camouflaged object detection, IEEE Transactions

on Information Forensics and Security 16 (2021) 5154–5166.
[44] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
[45] B. Jiang, L. Zhang, H. Lu, C. Yang, M.-H. Yang, Saliency detection via absorbing markov chain, in: Proceedings of the IEEE international conference on computer vision, 2013, pp. 1665–1672.
[46] Q. Yan, L. Xu, J. Shi, J. Jia, Hierarchical saliency detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2013, pp. 1155–1162.
[47] X. Li, H. Lu, L. Zhang, X. Ruan, M.-H. Yang, Saliency detection via dense and sparse reconstruction, in: Proceedings of the IEEE international conference on computer vision, 2013, pp. 2976–2983.
[48] W. Zhu, S. Liang, Y. Wei, J. Sun, Saliency optimization from robust background detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 2814–2821.
[49] C. Yang, L. Zhang, H. Lu, X. Ruan, M.-H. Yang, Saliency detection via graph-based manifold ranking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 3166–3173.
[50] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).
[51] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, H.-Y. Shum, Learning to detect a salient object, IEEE Transactions on Pattern analysis and machine intelligence 33 (2) (2011) 353–367.
[52] W. Zou, N. Komodakis, Harf: Hierarchy-associated rich features for salient object detection, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 406–414.
[53] Y. Li, X. Hou, C. Koch, J. M. Rehg, A. L. Yuille, The secrets of salient object segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 280–287.
[54] R. Achanta, S. Hemami, F. Estrada, S. Susstrunk, Frequency-tuned salient region detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1597–1604.
[55] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, A. Borji, Structure-measure: A new way to evaluate foreground maps, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4548–4557.
[56] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, A. Borji, Enhanced-alignment measure for binary foreground map evaluation, in: Proceedings of the International Joint Conference on Artificial Intelligence, 2018, pp. 698–704.
[57] Z. Wu, L. Su, Q. Huang, Cascaded partial decoder for fast and accurate salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3907–3916.
[58] N. Liu, J. Han, M.-H. Yang, Picanet: Learning pixel-wise contextual attention for saliency detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3089–3098.
[59] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, M. Jagersand, Basnet: Boundary-aware salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 7479–7489.
[60] H. Zhou, X. Xie, J.-H. Lai, Z. Chen, L. Yang, Interactive two-stream decoder for accurate and fast saliency detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 9141–9150.
[61] Z. Shao, J. Han, D. Marnerides, K. Debattista, Region-object relation-aware dense captioning via transformer. PP (2022).
[62] D. Liu, Y. Cui, W. Tan, Y. Chen, Sg-net: Spatial granularity network for one-stage video instance segmentation abs/2103.10284 (2021) 9811–9820.
[63] L. Li, T. Zhou, W. Wang, J. Li, Y. Yang, Deep hierarchical semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 1246–1257.
[64] Z. Wu, Y. Xiong, S. X. Yu, D. Lin, Unsupervised feature learning via non-parametric instance discrimination, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 3733–3742.
[65] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).
[66] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al., Segment anything, arXiv preprint arXiv:2304.02643 (2023).
[67] X. Wang, X. Zhang, Y. Cao, W. Wang, C. Shen, T. Huang, Seggpt: Segmenting everything in context, arXiv preprint arXiv:2304.03284 (2023).
[68] J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, Advances in Neural Information Processing Systems 33 (2020) 6840–6851.
[69] D. Liu, Y. Cui, L. Yan, C. Mousas, B. Yang, Y. Chen, Densernet: Weakly supervised visual localization using multi-scale feature aggregation 35 (2021) 6101–6109.
[70] J. Chen, J. Zhang, K. Debattista, J. Han, Semi-supervised unpaired medical image segmentation through task-affinity consistency. PP (2022) 1–1.
[71] D. Liu, Y. Cui, Y. Chen, J. Zhang, B. Fan, Video object detection for autonomous driving: Motion-aid feature calibration 409 (2020) 1–11.
[72] D. Yi, J. Su, W.-H. Chen, Probabilistic faster r-cnn with stochastic region proposing: Towards object detection and recognition in remote sensing imagery 459 (2021) 290–301.
[73] Y. Hua, D. Yi, Synthetic to realistic imbalanced domain adaption for urban scene perception 18 (2022) 3248–3255.