Weakly-supervised Salient Instance Detection

Xin Tian[12]
xtian@mail.dlut.edu.cn
Ke Xu[12]
kkangwing@mail.dlut.edu.cn
Xin Yang[1]
xinyang@dlut.edu.cn
Baocai Yin[13]
ybc@dlut.edu.cn
Rynson W.H. Lau[2]
rynson.lau@cityu.edu.hk

[1] Computer Science Department
Dalian University of Technology
Dalian, China

[2] Computer Science Department
City University of Hongkong
Hongkong SAR, China

[3] Pengcheng Lab
Shenzhen, China

Abstract

Existing salient instance detection (SID) methods typically learn from pixel-level annotated datasets. In this paper, we present the first weakly-supervised approach to the SID problem. Although weak supervision has been considered in general saliency detection, it is mainly based on using class labels for object localization. However, it is non-trivial to use only class labels to learn instance-aware saliency information, as salient instances with high semantic affinities may not be easily separated by the labels. We note that subitizing information provides an instant judgement on the number of salient items, which naturally relates to detecting salient instances and may help separate instances of the same class while grouping different parts of the same instance. Inspired by this insight, we propose to use class and subitizing labels as weak supervision for the SID problem. We propose a novel weakly-supervised network with three branches: a Saliency Detection Branch leveraging class consistency information to locate candidate objects; a Boundary Detection Branch exploiting class discrepancy information to delineate object boundaries; and a Centroid Detection Branch using subitizing information to detect salient instance centroids. This complementary information is further fused to produce salient instance maps. We conduct extensive experiments to demonstrate that the proposed method plays favorably against carefully designed baseline methods adapted from related tasks.

1 Introduction

Salient Object Detection (SOD) is a long-standing vision task that aims to segment visually salient objects in a scene. It often serves as a core step for downstream vision tasks like video object segmentation [38], object proposal generation [4], and image cropping [37]. Recent deep learning-based SOD methods have achieved a significant performance progress [16, 34, 35, 36, 40, 46, 50], benefited from the powerful representation learning capability of

neural networks and large-scale pixel-level annotated training data. Since annotating pixel-level labels is extremely tedious, there are some works [35, 43] that aim to explore cheaper image-level labels (*e.g.*, class labels) to train SOD models in a weakly-supervised manner.

Salient Instance Detection (SID) goes further from SOD as it is to identify each salient instance. This instance-level saliency information can further benefit vision tasks that requires fine-grained scene understanding, *e.g.*, image captioning [19], image editing [6] and semantic segmentation [10]. However, existing SID methods [11, 22, 44] still rely on large-scale annotated ground truth masks in order to learn how to segment salient instances with their boundaries delineated. Hence, it is worthwhile to study the SID problem from the weakly-supervised perceptive of using cheaper image-level labels.

A straightforward solution may be to use class labels to train a weakly-supervised SID model. However, using just class labels to learn a SID model is non-trivial for two reasons. First, class labels can help detect semantically predominant regions [47], but these regions are not guaranteed to be visually salient. Second, objects of the same class may not be easy distinguished due to their high semantic affinities. We observe that subitizing is naturally related to saliency instance detection. By predicting the number of salient objects, it can serve as global supervision that can help separate instances of the same class and cluster parts of an instance with diverse appearances into one.

Inspired by the above insight, we propose to learn a Weakly-supervised SID network (denoted WSID-Net) using class and subitizing labels. Our WSID-Net consists of three synergic branches: a salient object detection branch and a boundary detection branch are proposed to locate candidate salient objects and delineate their boundaries, by exploiting semantics from the class labels; a centroid detection branch is proposed to detect the centroid of each salient instance, by leveraging saliency cues from the subitizing labels. This information is fused to obtain the salient instance maps. To demonstrate the effectiveness of the proposed model, we compare it with a variety of baselines adapted from related tasks on the standard benchmark [22].

To summarize, this paper has three main contributions: 1) To the best of our knowledge, we propose the first weakly-supervised method for salient instance detection, which only requires image-level class and subitizing labels to obtain salient instance maps; 2) We propose a novel network (WSID-Net), with a novel centroid-based subitizing loss to exploit salient instance number information, and a novel Boundary Enhancement module to learn instance boundaries; 3) We conduct extensive experiments to analyze the proposed method, and verify its superiority against baselines adapted from related state-of-the-art approaches.

2 Related Work

Salient Instance Detection (SID). Existing SID methods are fully-supervised. Zhang *et al*. [44] propose to detect salient instances with bounding boxes, and propose a MAP-based optimization framework to regress a large amount of pre-defined bounding boxes into a compact number of instance-level bounding boxes of high confidences. However, their method based on bounding boxes cannot detect salient instances with accurately delineated boundaries. Other works predict pixel-wise masks for the detected salient instances, and typically rely on large amount of manually annotated ground truth labels. Specifically, Li *et al*. [22] propose to first predict the saliency mask and instance-aware saliency contour, and then use the existing Multi-scale Combinatorial Grouping (MCG) algorithm [5] to extract instance-level masks. Fan *et al*. [11] propose an end-to-end SID network based on the object detection model FPN [25], with a segmentation branch to segment the salient instances.

Unlike these existing SID methods, we propose in this paper to train a weakly-supervised network, which only requires two image-level labels, *i.e.*, the class and subitizing labels.

Salient Object Detection (SOD). SOD methods aim at generally detecting salient objects in a scene without differentiating the detected instances. Liu *et al*. [26] formulate the SOD task as a binary segmentation problem for segmenting out the visually conspicuous objects of an image via color and contrast histogram based priors. Traditional methods propose to leverage different hand-crafted priors to detect salient objects, *e.g.*, image colors and luminance [1], global and local contrast priors [7, 30], and background geometric distance prior [41]. Recently, deep learning based SOD methods achieve superior performance on standard SOD benchmarks [7, 17, 24, 32, 35, 41], by incorporating salient boundary knowledge [34, 40, 50], fusing deep features [16, 46], and designing attention mechanisms [36, 45]. Particularly, He *et al*. [15] propose to leverage numerical representation of subitizing to enrich spatial representations of salient objects. These methods are typically benefitted from the powerful learning ability of deep neural networks as well as large-scale annotated ground truth data. To alleviate the data annotation efforts, some methods [35, 43] propose to train weakly-supervised deep models using object class labels and class activation maps (CAMs) [47]. On the other hand, Li *et al*. [23] propose to leverage pre-trained contour network to generate pseudo labels for training the saliency detection network.

However, existing weakly-supervised SOD methods cannot be directly applied to our problem, as class labels cannot provide instance-level information. In this paper, we propose to use class and subitizing labels to train our SID model.

Weakly-supervised Semantic Instance Segmentation (SIS). SIS methods aim to detect all instances in a class-specific manner. Although they do not consider the saliency attribute of instances, they are related to our task as they try to segment the objects in an image into instances. Here, we briefly summarize latest weakly-supervised SIS methods, which are adopted as baseline methods for our task. Based on pixel affinities extracted from the class activation map, IRN [3] learns to predict object seeds and boundaries that can be used to infer the entire region of the target instance. PRM [48] first learns to predict peak response maps within class responses, where each peak is generally related to an instance. It then adopts off-the-shelf segment proposals [5] to obtain each instance based on the peaks. In comparison to PRM, PRM+D [8] further incorporates per-class object number information to learn better spatial distribution of peak-represented instances. Some other methods [21, 49] propose to refine the results of PRM [48] in an online way, via jointly learning from class labels and off-the-shelf segment proposals.

3 Methodology

Class labels are widely explored in weakly-supervised SOD methods for learning to localize candidate objects, based on the pixel-level semantic affinities derived from the network responses to the class labels. However, class labels lack instance-level information, causing over- and under-detection when salient instances are from the same category. We note that subitizing, a cheap image-level label that denotes the number of salient instances of a scene, can serve as a complementary supervision to the class labels. Hence, we propose to use both class and subitizing labels to address our weakly-supervised SID problem.

To this end, we propose a Weakly-supervised SID network (WSID-Net), as shown in Figure 1. WSID-Net has three branches: a *Salieny Detection Branch* for locating candidate salient objects; a *Centroid Detection Branch* for detecting the centroids of salient instances,

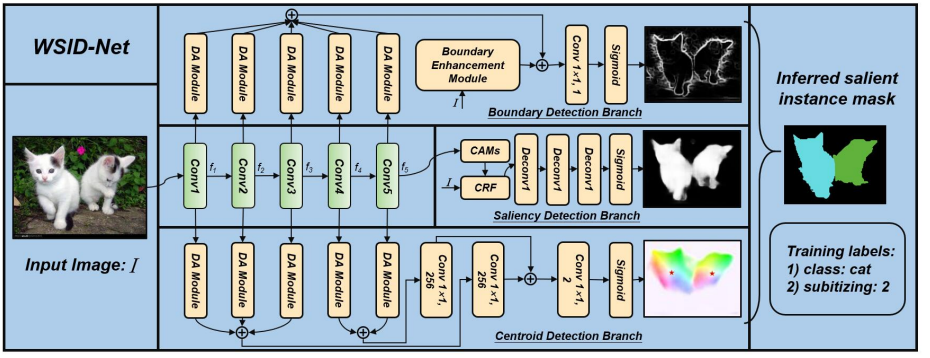

Figure 1: Pipeline overview. Our SID model is trained only using image-level class and subitizing labels. It has three synergic branches: (1) a Boundary Detection Branch for detecting object boundaries using class discrepancy information; (2) a Saliency Detection Branch for detecting objects using class consistency information; (3) a Centroid Detection Branch for detecting salient instance centroids using subitizing information. A random walk method is further applied to fuse these information to obtain final salient instance mask.

where subitizing knowledge is utilized in a novel loss function to provide regularization on the global number of instance centroids; and a *Boundary Detection Branch* for delineating salient instance boundaries, where a novel Boundary Enhancement (BE) module is introduced to resolve the discontinuity problem of detected boundaries. A novel Double Attention (DA) module is further incorporated to learn the context information for detecting centroids and boundaries.

3.1 Centroid Detection Branch

Detecting object centroids is crucial to separating the objects in a weakly-supervised scheme. Unlike existing semantic (instance) segmentation methods [3, 8, 21, 28, 48, 49] that detect the centroids based on network responses to the class labels, we propose to introduce subitizing information to explicitly supervise the salient centroid detection process.

Network structure. We adopt the image-to-image translation scheme, where our network outputs a 2D centroid map, of which the values of each pixel location indicate the offset vector to its instance centroid. The bottom part of Figure 1 shows the network structure of our centroid detection branch. Given an input image, we first extract multi-scale backbone features f_1 to f_5 and feed them to the DA modules for refinement (to be discussed in Section 3.3). The refined features are denoted as $f_1^{'}$ to $f_5^{'}$. We then fuse the high-level features to obtain f_h: $f_h = Conv(Concat(f_3^{'}, f_4^{'}, f_5^{'}))$, which is further fused with the low-level features to produce the centroid map $\mathcal{V}$: $\mathcal{V} = \sigma(Conv(Conv(Concat(f_h, f_1^{'}, f_2^{'}))))$.

Centroid-based Subitizing loss. It has been shown that penalizing the centroid loss [3, 28] helps cluster local pixels with high semantic affinities. However, it typically fails when salient instances from the same object category have varying shapes and appearances. The reason is that the clustering process of local pixels lacks global saliency supervision on it. Hence, we introduce the centroid-based subitizing loss $\mathcal{L}_{\mathcal{SU}}$ to resolve this problem. We use subitizing to explicitly supervise the number of predicted centroids, which helps constrain the pixel clustering process. We use the Mean Square Error (MSE) to measure $\mathcal{L}_{\mathcal{SU}}$:

$$\mathcal{L}_{\mathcal{SU}} = MSE(t_{\prod_{\mathcal{V}(x_i)} x_i \in \mathcal{S}}, t^*), \tag{1}$$

where t^* is the subitizing information, $\mathcal{S}$ denotes the predicted saliency region. $\prod_{\mathcal{V}(x_i)} x_i \in \mathcal{S}$ denotes the predicted offset vectors in the saliency region. $t_{\prod_{\mathcal{V}(x_i)} x_i \in \mathcal{S}}$ denotes the number of predicted centroid extracted from the offset vectors of the pixels in the saliency region. The loss $\mathcal{L}_{\mathcal{SU}}$ only backpropagates to update the offset vectors in the saliency region, avoiding the learning process of instance centroid detection being distracted by the non-salient background.

Figure 2 visualizes the results from centroid detection and the corresponding instance segmentation, with and without using the centroid-based subitizing $\mathcal{L}_{\mathcal{SU}}$ loss function. We can see that the network groups the two dogs into one when not using $\mathcal{L}_{\mathcal{SU}}$, as these two dogs have similar appearances and lie next to each other (columns 3 and 4). By introducing $\mathcal{L}_{\mathcal{SU}}$, the network is able to predict a correct number of centroids, and generate reasonable salient instance masks compared with the ground truth (columns 5 and 6).

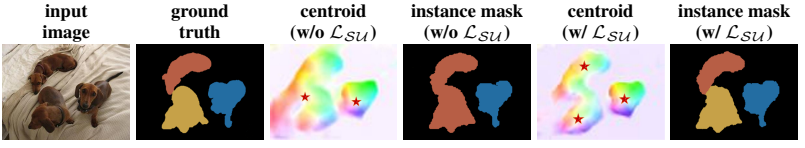

Figure 2: Visualization of the centroid detection branch with and without $\mathcal{L}_{\mathcal{SU}}$.

3.2 Boundary Detection Branch

Boundaries provide strong cues for separating salient instances. Unlike fully-supervised SID methods that learn boundary-aware information based on pixel-level ground truth masks, we propose the Boundary Enhancement module to leverage the Canny prior [18] to delineate continuous instance boundaries.

Network structure. The top part of Figure 1 shows the architecture of the boundary detection branch. Given an input image $\mathcal{I}$, we obtain refined backbone features ($f_1^{'}$ to $f_5^{'}$) using DA modules (to be discussed in Section 3.3) before they are concatenated and computed to predict the boundary map. We also feed the input image into the BE module to obtain enhanced edge features f_b. The output boundary map $\mathcal{B}$ is then computed as: $\mathcal{B} = \sigma(Conv(Concat(f_1^{'}, ..., f_5^{'}, f_b)))$, where σ is the sigmoid activation function.

BE module. We apply a random walk algorithm to search a salient instance from a centroid to its boundary. However, it may fail when part of the boundary is discontinuous as the random walk algorithm will also search the region outside the boundary. Hence, we propose the BE module to incorporate the edge prior for learning continuous instance boundaries, as shown in Figure 3. Specifically, we first extract low-level features along the horizontal and vertical directions from the input image, by two 1×7 and 7×1 convolution layers. These low-level features are then fed into three Residual Blocks [14] for feature refinement, which are further concatenated with enriched edges computed from the Canny operator [18]. To compute the final enriched boundary features, another 1×1 convolution layer is applied.

Figure 4 visualizes two examples of boundary detection and the corresponding salient instance detection with and without the BE module. We can see that our BE module helps detect the boundaries between objects, which is crucial to salient instance segmentation.

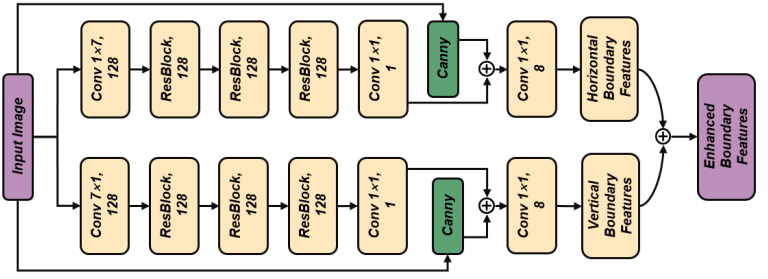

Figure 3: Illustration of our novel Boundary Enhancement module.

Figure 4: Visualization of the boundary detection branch with and without the BE module.

3.3 Double Attention (DA) Module

Detecting instance centroids and boundaries are two highly coupled sub-tasks, *i.e.*, they can influence each other and further affect the SID performance. To efficiently learn these two sub-tasks, we propose the Double Attention (DA) module. Its design is based on two observations. First, since salient instances may have various shapes, we thus need to capture long-range spatial contextual information. Second, cross-class ambiguities of pseudo affinity labels influence both sub-tasks, while the class information from the channel-wise contexts can help address this problem. Hence, we combine channel-wise and spatial-wise attention mechanisms and organize them in parallel to form our DA module. We apply the DA module to both the centroid detection and the boundary detection branches, and share their weights. Unlike existing dual attention mechanisms [12, 39] that are only used to enhance the feature discriminatively, our DA module also allows information exchanges across these two branches, resulting in an improvement on both sub-tasks.

Figure 5 shows the structure of our DA module. The top and bottom branches are channel-wise and spatial-wise attention blocks, respectively. Specifically, given the input features f_n, we compute the channel-wise attention features $\mathcal{F}_c$ as: $\mathcal{F}_c = \sigma(MLP(AvgPool_c(f_n)) + MLP(MaxPool_c(f_n)))$, where $MaxPool_c$ and $AvgPool_c$ denote two channel-wise pooling operations, and MLP is the multi-layer perception with one hidden layer to generate the attention features. We also compute the spatial-wise attention features $\mathcal{F}_s$ as: $\mathcal{F}_s = \sigma(Conv_{7\times7}([AvgPool_s(f_n); MaxPool_s(f_n)]))$, where $Conv_{7\times7}$ is a convolutional layer with kernel size 7. The final attention features $f_n^{'}$ are then computed as: $f_n^{'} = f_n \times \mathcal{F}_c + f_n \times \mathcal{F}_s$, where $\times$ denotes the dot product operation, and $+$ is the element-wise summation operation.

Figure 6 shows the effectiveness of the proposed Double Attention module in enhancing the boundary and centroid detection performances.

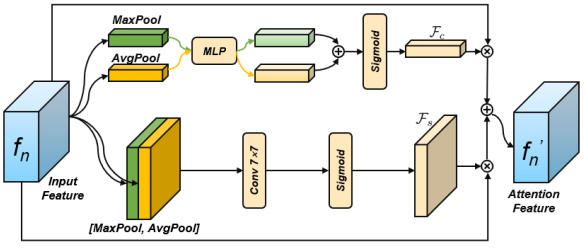

Figure 5: Double Attention module.

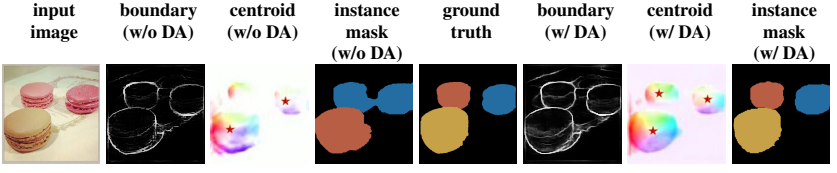

Figure 6: Visualization of the DA module with and without the DA module.

4 Experiments

4.1 Training and evaluation details

Datasets and metric. Our full network is trained on two image-level labels, class and subitizing. We use PASCAL VOC 2012 [9], which is a semantic and instance segmentation dataset. However, we only use its class labels to train our network. The training set contains a total of 10,582 images of 20 classes. ILSO [22] is a SID dataset. It contains 500 images with instance-aware pixel labels for training. For our weakly-supervised training, we extract the numbers of salient instances of these images as our subitizing labels. We perform evaluations on the test set of ILSO [22], which has 300 images with ground truth instance masks. We use the mean Average Precision (mAP) metric [13] to evaluate the SID performance.

Training and Inference. We train the proposed network separately. We train the centroid detection branch using the proposed centroid-based subitizing loss together with the centroid loss introduced in [3, 28]. We train the boundary detection branch using the boundary loss introduced in [2, 3]. To train the saliency detection branch, we follow existing weakly-supervised SOD methods to use pseudo masks derived from class labels. Specifically, we first compute class activation maps via [47]. We then feed these maps together with the input image to a Conditional Random Field [20] to generate pseudo object maps, and use these pixel-level pseudo labels to train the saliency detection branch. During inference, given an input image, WSID-Net first computes the centroids, boundaries, and saliency maps. To segment each salient instance, a random walk algorithm is used to detect salient regions, starting from the detected centroids until reaching the boundaries.

4.2 Implementation details

We implement WSID-Net on the Pytorch framework [29]. Both training and testing are performed on a PC with an i7 4GHz CPU and a GTX 1080Ti GPU. CRF is used to generate or refine pseudo labels. The hyper parameters of CRF are set as $w_1 = 4.0$, $w_2 = 3.0$, $\sigma_\alpha = 49.0$, $\sigma_\beta = 5.0$, and $\sigma_\gamma = 3.0$. We choose ResNet50 as the backbone for all three branches in WSID-Net. The backbone is initialized as in [33]. Input images are resized to 512×512

resolution. To minimize the loss function, we use the SGD optimizer with batch size 6 and initial learning rate 0.01. The learning rate decreases following poly policy ($lr_{itr} = lr_{init}(1 - \frac{itr}{max_{itr}})^{\gamma}$). We train our WSID-Net for 5 epoches.

4.3 Comparion with the State-of-the-art Methods

As we are the first to propose a weakly-supervised SID method, we compare our method to 2 existing fully-supervised state-of-the-art SID methods: S4Net [11] and MSRNet [22]. We also prepare the following baselines from related tasks for evaluation. We choose 6 state-of-the-art weakly-supervised methods, with two from the SOD task C2SNet [23] and NLDF [27]; one from the SID task MAP [44]; one from the object detection (OD) task, DeepMask [31]; and two from the Semantic Instance Segmentation task, PRM+D [8] and IRN [3]. We adapt them by adding different post-processing strategies to these methods for deriving instance-level saliency maps from their original outputs, or modifying their networks and retrain them using our training data. Details are summarized as follows:

- C2SNet [23] and NLDF [27] are proposed for salient object detection with contour prediction. We apply the MCG method [5], which takes a contour map as input and outputs segment proposals, to obtain multiple salient instance proposals, and then use MAP [44] to filter out proposals with low confidences.
- MAP [44] is a fully-supervised SID method, which learns to predict the bounding boxes of salient instances. Since it cannot output salient instance masks, we feed both the image and the bounding box to a CRF [20] to obtain the segmented masks.
- DeepMask [31] learns to predict class-agnostic segment proposals with object scores. We utilize a weakly-supervised SOD method WSS [35] to filter out non-salient segment proposals by calculating the IoU between the object mask and the salient mask, and set the IoU threshold to 0.75.
- IRN [3] learns to predict class-specific segment proposals. We utilize the same filtering method as in DeepMask to select salient instances.
- PRM+D [8] is trained with class and per-class subitizing labels. We add one additional convolutional layer at the end of the network to merge their per-class outputs (originally 20 output maps for 20 classes) into one class-agnostic map, and then retrain it using our training data.

4.4 Performance Evaluation

Quantitative evaluation. We quantitatively evaluate our method in Table 1†. mAP@0.7 is the most difficult metric as it requires the IoU value to be over 70%. Our method achieves a better performance of about 10% over the second-place weakly-supervised baseline. These results show that our method achieves the best performance using just two types of image-level labels.

Qualitative evaluation. We further qualitatively compare our method as shown in Figure 7. Our method is able to delineate the instance boundaries clearly, and output an accurate number of segmented salient instances directly, as shown in column 9. In contrast, (1) PRM+D

† As of today, the codes for MSRNet [22] are still not available. Following [11], we directly copy the numbers reported in [22] to our submission for a quantitative comparison.

and IRN fail to detect integral instances with inferior detected boundaries (*e.g.*, rows 1 and 9); (2) C2SNet and NLDF tend to recognize texture boundaries, resulting in fragmented instances (*e.g.*, rows 2, 3 and 4); (3) DeepMask and S4Net suffer from the over-detection problem, as they fail to distinguish instance proposals belonging to the same instance (*e.g.*, row 2); (4) and MAP is a bounding-box based method that fails to get clear instance boundary even post-processed by a widely adopted segmentation method, CRF (*e.g.*, rows 1 and 2). Overall, our method outperforms the baselines, as a result of the centroid-based subitizing loss and the carefully designed BE and DA modules.

Methods	Original task	Supervision types	Training labels	Auxiliary models	mAP @0.5↑	mAP @0.7↑
MSRNet [22]	SID	FS	object-level and instance-level pixel masks	MAP [44], MCG [5]	65.3%	52.3%
MAP [44]	SID	FS	instance-level bounding box	N/A	56.6%	24.8%
S4Net [11]	SID	FS	instance-level pixel mask	N/A	82.2%	59.6%
C2SNet [23]	SOD	WS	unlabeled images	CEDN [42], MAP [44], MCG [5]	41.1%	25.4%
NLDF [27]	SOD	WS	object-level pixel mask	MAP [44], MCG [5]	45.5%	24.5%
DeepMask [31]	OD	WS	instance-level bounding box	N/A	37.1%	20.5%
PRM+D [8]	SIS	WS	class, subitizing labels	MCG [5]	49.6%	31.2%
IRN [3]	SIS	WS	class label	N/A	57.1%	37.4%
Ours	SID	WS	class, subitizing labels	N/A	**61.9%**	**47.2%**

Table 1: Quantitative evaluation of our method against six baseline methods and state-of-the-art fully-supervised SID methods. For the compared methods, we show their original tasks, supervision types, training labels and auxiliary pre-trained models in the 2*nd* to 5*th* columns. SID, SOD, OD, SIS are short of salient instance detection, salient object detection, object detection and semantic instance segmentation, respectively. FS and WS denote Fully-Supervised and Weakly-Supervised. Best performances among the weakly-supervised methods are marked in **red**.

4.5 Internal Analysis

We first investigate how our BE, DA modules, and $\mathcal{L}_{\mathcal{SU}}$ affect the SID performance. We provide ablation study on our model design based on the mAP@IoU metric. From the results in Table 2, we can see that the SID performance is continuously increased as we incorporate these modules. This shows that these modules can help boost the performances of the centroid and boundary detection sub-tasks, which play a vital role in detecting salient instances. Figures 2, 4, and 6 provide additional visual comparisons to demonstrate the effectiveness of the BE, DA modules and the $\mathcal{L}_{\mathcal{SU}}$.

We also evaluate the design choices of the DA module and the influence of using different backbones. Due to page limitation, we show these analytical results in the Supplemental.

5 Conclusion

In this paper, we propose the first weakly-supervised SID method that is trained on class and subitizing labels. Our WSID-Net learns to predict object boundary, instance centroid, and salient region. By using the proposed Boundary Enhancement module, Double Attention module, and centroid-based subitizing loss, our method can identify and segment each

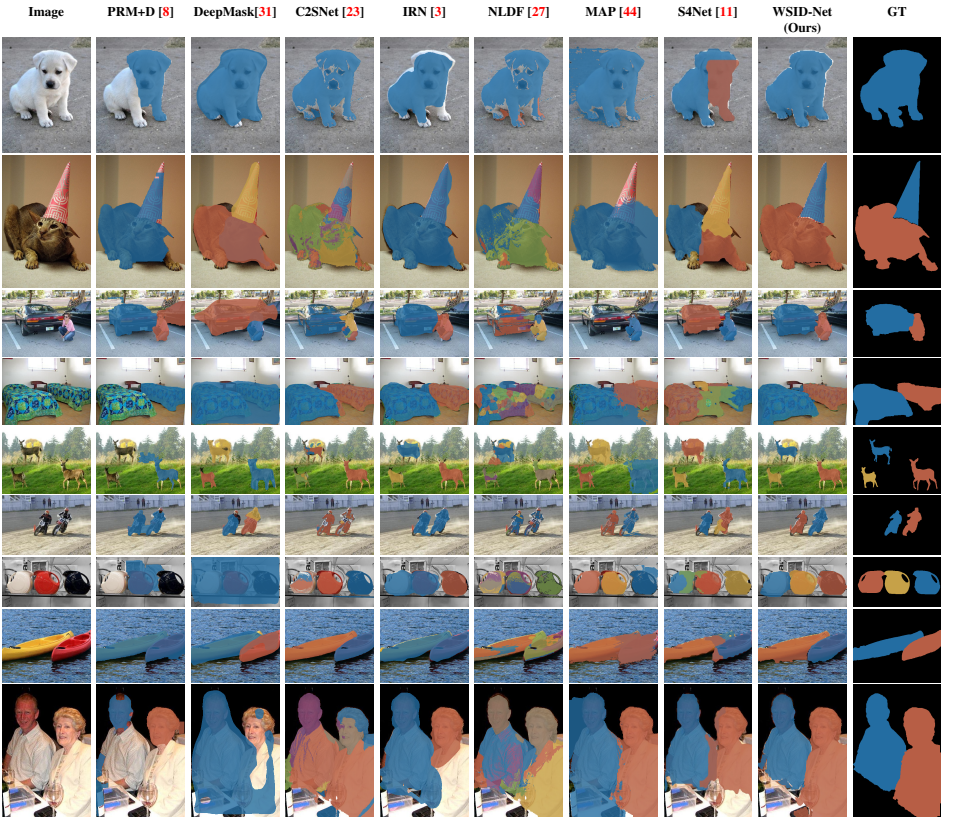

Figure 7: Qualitative results of our method, compared with existing fully-supervised methods (S4Net[11] and MAP [44]) and modified baselines (PRM+D [8], DeepMask [31], C2SNet [23], NLDF [27], and IRN [3]). Refer to Section 4.3 and Table 1 on how we modify and train these baselines, in order to perform appropriate comparison.

salient instance effectively. Both quantitative and qualitative experiments demonstrate the effectiveness of the proposed method compared with baseline methods.

Our method does have limitations. It may fail when our saliency detection branch (as well as existing weakly-supervised SOD methods) cannot detect the majority of the salient regions, due to complex background textures and colors, as shown in Figure 8. As a future work, we are exploring to incorporate a discriminative network of generative adversarial learning to improve the SOD performance of our saliency detection branch, and extend our method to handle videos.

method	mAP@0.5↑	mAP@0.7↑
Ours (w/o DA, BE, $\mathcal{L}_{\mathcal{SU}}$)	57.1%	37.4%
Ours (w/o DA)	60.3%	45.1%
Ours (w/o BE)	58.2%	44.3%
Ours (w/o $\mathcal{L}_{\mathcal{SU}}$)	59.9%	45.0%
Ours	**61.9%**	**47.2%**

Table 2: Ablation study of WSID-Net.

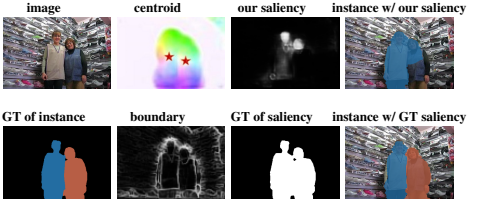

Figure 8: A failure case.

References

[1] Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Süsstrunk. Frequency-tuned salient region detection. In *CVPR*, 2009.

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, 2018.

[3] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *CVPR*, 2019.

[4] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *IEEE PAMI*, 2012.

[5] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *CVPR*, 2014.

[6] Tao Chen, Ming-Ming Cheng, Ping Tan, Ariel Shamir, and Shi-Min Hu. Sketch2photo: Internet image montage. In *ACM TOG*, 2009.

[7] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE TPAMI*, 2014.

[8] Hisham Cholakkal, Guolei Sun, Fahad Shahbaz Khan, and Ling Shao. Object counting and instance segmentation with image-level supervision. In *CVPR*, 2019.

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010.

[10] Ruochen Fan, Qibin Hou, Ming-Ming Cheng, Gang Yu, Ralph R Martin, and Shi-Min Hu. Associating inter-image salient instances for weakly supervised semantic segmentation. In *ECCV*, 2018.

[11] Ruochen Fan, Ming-Ming Cheng, Qibin Hou, Tai-Jiang Mu, Jingdong Wang, and Shi-Min Hu. S4net: Single stage salient-instance segmentation. In *CVPR*, 2019.

[12] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *CVPR*, 2019.

[13] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *ECCV*, 2014.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[15] Shengfeng He, Jianbo Jiao, Xiaodan Zhang, Guoqiang Han, and Rynson WH Lau. Delving into salient object subitizing and detection. In *ICCV*, 2017.

[16] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip HS Torr. Deeply supervised salient object detection with short connections. In *CVPR*, 2017.

[17] Huaizu Jiang, Jingdong Wang, Zejian Yuan, Yang Wu, Nanning Zheng, and Shipeng Li. Salient object detection: A discriminative regional feature integration approach. In *CVPR*, 2013.

[18] Canny John. A computational approach to edge detection. *IEEE TPAMI*, 1986.

[19] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.

[20] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NeurIPS*, 2011.

[21] Issam H Laradji, David Vazquez, and Mark Schmidt. Where are the masks: Instance segmentation with image-level supervision. In *BMVC*, 2019.

[22] Guanbin Li, Yuan Xie, Liang Lin, and Yizhou Yu. Instance-level salient object segmentation. In *CVPR*, 2017.

[23] Xin Li, Fan Yang, Hong Cheng, Wei Liu, and Dinggang Shen. Contour knowledge transfer for salient object detection. In *ECCV*, 2018.

[24] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *CVPR*, 2014.

[25] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

[26] Tie Liu, Zejian Yuan, Jian Sun, Jingdong Wang, Nanning Zheng, Xiaoou Tang, and Heung-Yeung Shum. Learning to detect a salient object. In *CVPR*, 2007.

[27] Zhiming Luo, Akshaya Mishra, Andrew Achkar, Justin Eichel, Shaozi Li, and Pierre-Marc Jodoin. Non-local deep features for salient object detection. In *CVPR*, 2017.

[28] Davy Neven, Bert De Brabandere, Marc Proesmans, and Luc Van Gool. Instance segmentation by jointly optimizing spatial embeddings and clustering bandwidth. In *CVPR*, 2019.

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

[30] Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. Saliency filters: Contrast based filtering for salient region detection. In *CVPR*, 2012.

[31] Pedro OO Pinheiro, Ronan Collobert, and Piotr Dollár. Learning to segment object candidates. In *NeurIPS*, 2015.

[32] Jianping Shi, Qiong Yan, Li Xu, and Jiaya Jia. Hierarchical image saliency detection on extended cssd. *IEEE TPAMI*, 2015.

[33] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014.

[34] Jinming Su, Jia Li, Yu Zhang, Changqun Xia, and Yonghong Tian. Selectivity or invariance: Boundary-aware salient object detection. In *ICCV*, 2019.

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

[36] Tiantian Wang, Lihe Zhang, Shuo Wang, Huchuan Lu, Gang Yang, Xiang Ruan, and Ali Borji. Detect globally, refine locally: A novel approach to saliency detection. In *CVPR*, 2018.

[37] Wenguan Wang and Jianbing Shen. Deep cropping via attention box prediction and aesthetics assessment. In *ICCV*, 2017.

[38] Wenguan Wang, Jianbing Shen, and Fatih Porikli. Saliency-aware geodesic video object segmentation. In *CVPR*, 2015.

[39] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, 2018.

[40] Yingyue Xu, Dan Xu, Xiaopeng Hong, Wanli Ouyang, Rongrong Ji, Min Xu, and Guoying Zhao. Structured modeling of joint deep feature and prediction refinement for salient object detection. In *ICCV*, 2019.

[41] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, 2013.

[42] Jimei Yang, Brian Price, Scott Cohen, Honglak Lee, and Ming-Hsuan Yang. Object contour detection with a fully convolutional encoder-decoder network. In *CVPR*, 2016.

[43] Yu Zeng, Yunzhi Zhuge, Huchuan Lu, Lihe Zhang, Mingyang Qian, and Yizhou Yu. Multi-source weak supervision for saliency detection. In *CVPR*, 2019.

[44] Jianming Zhang, Stan Sclaroff, Zhe Lin, Xiaohui Shen, Brian Price, and Radomir Mech. Unconstrained salient object detection via proposal subset optimization. In *CVPR*, 2016.

[45] Lu Zhang, Ju Dai, Huchuan Lu, You He, and Gang Wang. A bi-directional message passing model for salient object detection. In *CVPR*, 2018.

[46] Ting Zhao and Xiangqian Wu. Pyramid feature attention network for saliency detection. In *CVPR*, 2019.

[47] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016.

[48] Yanzhao Zhou, Yi Zhu, Qixiang Ye, Qiang Qiu, and Jianbin Jiao. Weakly supervised instance segmentation using class peak response. In *CVPR*, 2018.

[49] Yi Zhu, Yanzhao Zhou, Huijuan Xu, Qixiang Ye, David Doermann, and Jianbin Jiao. Learning instance activation maps for weakly supervised instance segmentation. In *CVPR*, pages 3116–3125, 2019.

[50] Yunzhi Zhuge, Gang Yang, Pingping Zhang, and Huchuan Lu. Boundary-guided feature aggregation network for salient object detection. *IEEE Signal Processing Letters*, 2018.