
Few-Cost Salient Object Detection with Adversarial-Paced Learning

Dingwen Zhang¹, Haibin Tian¹ and Jungong Han^{2✉}

¹School of Mechano-Electronic Engineering, Xidian University, Xi'an, Shaanxi 710071

²Computer Science Department, Aberystwyth University, Ceredigion, SY23 3FL
zdw@xidian.edu.cn, haibintian@foxmail.com, jungonghan77@gmail.com

Abstract

Detecting and segmenting salient objects from given image scenes has received great attention in recent years. A fundamental challenge in training the existing deep saliency detection models is the requirement of large amounts of annotated data. While gathering large quantities of training data becomes cheap and easy, annotating the data is an expensive process in terms of time, labor and human expertise. To address this problem, this paper proposes to learn the effective salient object detection model based on the manual annotation on a few training images only, thus dramatically alleviating human labor in training models. To this end, we name this task as the few-cost salient object detection and propose an adversarial-paced learning (APL)-based framework to facilitate the few-cost learning scenario. Essentially, APL is derived from the self-paced learning (SPL) regime but it infers the robust learning pace through the data-driven adversarial learning mechanism rather than the heuristic design of the learning regularizer. Comprehensive experiments on four widely-used benchmark datasets demonstrate that the proposed method can effectively approach to the existing supervised deep salient object detection models with only 1k human-annotated training images. The project page is available at <https://github.com/hb-stone/FC-SOD>.

1 Introduction

With the goal of automatically discovering object regions that attract human attention from the given image scenes, salient object detection has become prevalent in the computer vision community [1–3]. Due to its wide range of applications, large amounts of efforts have been made to build powerful deep convolutional network models for addressing this problem. Relying on the large-scale human-annotated training image data, methods presented in recent years have been undergoing unprecedentedly rapid development. However, as it is often time-consuming and expensive to provide the manually annotated pixel-wise ground-truth annotation, the training processes of the most existing methods are costly in terms of time and money. To this end, this paper studies the challenge to learn an effective salient object detection model by only using the manual annotation on a few training images.

Inspired by the few-shot learning problems [4, 5] that use only a few training samples of the targets, we name the investigated problem as the few-cost salient object detection (FC-SOD) problem as it costs only a few annotated training samples. More specifically, in FC-SOD, the scenario is that the training data contain the large scale training images, but only a few of them have the pixel-wise annotation on the salient objects. Such a problem sounds also similar to a semi-supervised learning problem. As [6] has defined the semi-supervised SOD (SS-SOD) task as the task to partially label the regions within each image firstly and then use both the labeled and unlabeled regions to learn the saliency model, we define the task considered in this work as FC-SOD to avoid confusion.

When designing the few-cost learning framework for salient object detection, the key problem is to progressively annotate the unlabeled training images according to the knowledge mined from the small scale annotated training images. However, such a learning procedure may turn to trivial solutions when noisy or wrong annotations are set to the unlabeled training images and introduced into the intermediate learning process. A simple yet effective way to alleviate this problem is the use of the self-paced learning (SPL) mechanism [7] in the few-cost learning framework as SPL is inherently a robust learning mechanism that helps the learning system explore the samples containing truthful knowledge (i.e., those with accurate labels) while screening the samples with unreliable knowledge (i.e., those with noisy or wrong labels). Such robust learning capacity has also been proved by recent studies in vision tasks [8–12]. For example, [13, 14] apply the SPL process to refine the saliency maps obtained in co-saliency detection. [15, 16] integrate SPL and adversarial learning for domain adaption and clustering, respectively.

At the core of SPL is the design of the self-paced regularizer, based on which the learner can dynamically assign proper learning weights to the samples—reliable labels are assigned with large learning weights while noisy labels are attached with small learning weights—and this dynamical weighting process leads to the robust learning pace to guide the learning procedure. Currently, the main strategy of SPL methods designs the self-paced regularizer based on human knowledge in the corresponding task domain. This strategy is, to some extent, heuristic and may lead to suboptimal solutions due to the insufficient exploration of the data. Thus, a more reasonable way to infer the robust learning pace might be in a data-driven manner, where the concrete formulation of the learning weights is learned from the data rather than being manually designed by humans. In this way, the learning system can, on one hand, alleviate its dependency on manual design, which endows stronger learning capacity to the learner. On the other hand, by leveraging the samples from the corresponding task domain, it can obtain the more suitable learning pace for any task under investigation.

To implement such a learning mechanism, we propose a novel adversarial-paced learning framework, which is derived from the conventional SPL and driven by the underlying relationship between the optimization processes of SPL and the well-known generative adversarial learning (GAL) [17]. Specifically, it is known that the alternative optimization strategy commonly used to solve the SPL problem can be considered as the majorization minimization algorithm that is implemented on a latent SPL objective function [18]. While the optimization process of GAL is also a min-max game. Thus, the proposed APL framework can be implemented with a similar optimization process as the conventional SPL methods but with a different data-driven mechanism to infer the learning weights and generate the learning pace. In APL, the adversarial learning mechanism will enable the learner to tell which of the predicted labels are “real”, i.e., reliable, while which are not.

To sum up, this work mainly contains the following three-fold contributions:

- We explore an under-studied task called few-cost salient object detection. Compared with the conventional fully supervised salient object detection, it requires only the manual annotation on a small number of training images and thus can alleviate the annotation cost for training deep salient object detectors.
- We reveal the underlying relationship between SPL and GAL to establish a novel adversarial-paced learning framework. By implicitly encoding the pace regularizer in an additional model called pace-generator, APL can infer the robust learning pace through a data-driven adversarial learning mechanism rather than the heuristic design of the learning regularizer.
- Comprehensive experiments on widely-used benchmark datasets have been implemented to evaluate the effectiveness of the proposed approach. Notably, by using the annotation of only 1k training images, the proposed approach outperforms the existing un-/semi-/weakly supervised SOD approaches and performs comparably to the fully supervised SOD models.

2 Previous Works

Salient Object Detection. In light of the advanced development in deep learning, recent salient object detection methods mainly adopted the CNN models to learn saliency patterns under a fully supervised fashion. Most of these methods, e.g., [19–24], focus on extracting representative deep features in more effective and efficient ways. For learning strong feature representations, a new trend in this field is appeared, which provides richer supervision to guide the network learning process. Under this strategy, some recent works introduce the human-annotated contour information into

the network learning process. For example, Wu et al. [25] integrated salient object detection and foreground contour detection tasks in an intertwined manner, which enables the learned model to generate saliency maps with uniform highlight regions. Liu et al. [26] built simple yet effective pooling-based modules to decode the deep features to infer both saliency maps and contour maps.

Different from the above-mentioned direction, this work explores an alternative direction to advance the SOD community—Instead of acquiring richer supervision, this work studies how to shrink the supervision. Research from this direction could dramatically reduce the labor costs and endow the deep model stronger learning capacity. Notice that there is also a small number of works [27–29] that share the similar spirit with this work. However, the problems considered by them are under the weakly supervised or unsupervised learning scenarios, which are distinct from our investigated few-cost learning problem. Unlike our work, Yan et al. [30] define their problem on sparsely annotated video frames. They train video salient object detector by leveraging the dependencies among adjacent video frames. Notice that the upper bound of FCSOD should be higher than unsupervised SOD as unsupervised SOD does not use any human annotation. With the same label cost, the upper bound of FCSOD and weakly supervised SOD (WSSOD) should be close. However, as FCSOD leverages a small number of strong annotations while WSSOD leverages a lot of weak annotations, FCSOD should theoretically work better when dealing with data with small domain shifts.

Semi-supervised GAL. The proposed APL is also related to the semi-supervised GAL (SS-GAL) framework, such as [31–33], as both of them use the labeled data and unlabeled data in the learning procedure. However, the generators and discriminators used in APL and SS-GAL play very distinct functions—In SS-GAL, the generator is used to generate extra training samples from the input noise signals while the discriminator acts as a multi-class classifier to predict labels for the input training samples. In contrast, the generator in APL is used to predict labels for the input training samples while the discriminator is used to judge whether the input label is with a realistic structure. The core difference between APL and SS-GAL is that the unlabeled data in APL are used to learn the inference function for label weighting whereas the unlabeled data in SS-GAL are used to learn the mapping function for feature representation. The work of [34] is also related to this work, where an IoT-oriented saliency learning framework is presented with the intention to leverage both labeled and unlabeled data from different problem domains for training. In contrast, our work aims to minimize the annotation cost for saliency learning in a single domain.

Another interesting SS-GAL framework is proposed by [35], which is intuitively analogous with our approach. However, from the perspective of the high-level idea, our work differs from [35] by deriving APL from the SPL regime and establishing the learning framework in a theoretically-sound manner rather than a heuristic manner. While from the perspective of implementation details, our works differ from [35] by proposing a novel global structure-guided pixel weighting scheme and designing the different objective function and optimization strategy. By revealing the underlying relationship between SPL and GAL mechanism and explicitly modeling the reliability weights \mathbf{V} (see Eq.2), this work could provide a theoretical explanation and a new interpretation of the learning framework of [35].

It is also worth mentioning that when comparing the semi-supervised semantic segmentation (SSS) problem and the investigated few-cost salient object detection problem, besides the superficial difference in the number of classes, the challenges met by them are also different. Specifically, as the salient class that needs to be segmented in FCSOD would usually cover a number of different semantics rather than forming by a specific semantic, FCSOD is encountered with heavier intra-class variance than SSS. This would bring the challenging learning ambiguity issue in FCSOD. Besides, the current SSS methods usually leverage the semantic vector as an informative attribute to guide the GAN-based semi-supervised learning process, which, however, is absent from the FCSOD task. Consequently, such SSS algorithms could not be easily applied to FCSOD.

3 Adversarial-Paced Learning

Formulation. Given a small-scale collection \mathcal{D}^l , which consists of the manually labeled training images $\{\mathbf{X}_i^l, \mathbf{Y}_i^l\}$, and a larger-scale collection \mathcal{D}^u that contains the unlabeled training images $\{\mathbf{X}_j^u\}$, we denote the pseudo label of \mathbf{X}_j^u denotes as \mathbf{Y}_j^u , which needs to be inferred during the learning process. To solve this problem, the existing methods may adopt the SPL [7]-based learning framework to infer the pseudo labels for the unlabeled images and then involve the self-paced regularizer to guide a robust learning procedure to against the noise brought by the inaccurate pseudo label. Such

Algorithm 1 The conventional SPL scheme. $g(\cdot)$ and $f(\cdot)$ denote the learning model and the self-paced regularizer, respectively. Notice that we only show the main objective function and ignore the constraints here.

Input: The set of labeled samples $\{\mathbf{X}_i^l, \mathbf{Y}_i^l\}$; The set of unlabelled samples \mathbf{X}_i^u ;
Output: Model parameter Θ ;
1: Given objective function:
2: $\min_{\Theta, \mathcal{V}, \mathcal{Y}^u} \mathcal{L}^l(\Theta) + \mathcal{L}^u(\mathcal{Y}^u, \Theta, \mathcal{V}) + \lambda f(\mathcal{V})$
3: Initializing Θ and \mathcal{V} ;
4: **repeat**
5: Update \mathcal{Y}^u with fixed Θ and \mathcal{V} ;
6: Calculate $g(\mathbf{X}_j^u|\Theta)$
7: Update \mathcal{V} with fixed Θ and \mathcal{Y}^u ;
8: Optimize $\min_{\mathcal{V}} \mathcal{L}^u(\mathcal{Y}^u, \Theta, \mathcal{V}) + \lambda f(\mathcal{V})$
9: Update Θ with fixed \mathcal{Y}^u and \mathcal{V} ;
10: Optimize $\min_{\Theta} \mathcal{L}^l(\Theta) + \mathcal{L}^u(\mathcal{Y}^u, \Theta, \mathcal{V})$
11: **until** converge
12: **return** Model parameter Θ ;

Algorithm 2 The proposed APL scheme. $T(\cdot)$ and $P(\cdot)$ denote the task-predictor and the pace-generator, respectively. Notice that we only show the main objective function and ignore the constraints here.

Input: The set of labeled samples $\{\mathbf{X}_i^l, \mathbf{Y}_i^l\}$; The set of unlabelled samples \mathbf{X}_i^u ;
Output: Model parameter $\Theta = \{\Psi, \Phi\}$;
1: Given objective function:
2: $\min_{\Psi, \mathcal{V}, \mathcal{Y}^u} \max_{\Phi} \mathcal{L}^l(\Psi) + \mathcal{L}^u(\mathcal{Y}^u, \Psi, \mathcal{V}) + \beta \mathcal{L}^p(\mathcal{Y}^u, \Psi, \Phi)$
3: Initializing Θ and \mathcal{V} ;
4: **repeat**
5: Update \mathcal{Y}^u with fixed $\Theta = \{\Psi, \Phi\}$ and \mathcal{V} ;
6: Calculate $T(\mathbf{X}_j^u|\Psi)$
7: Update \mathcal{V} with fixed $\Theta = \{\Psi, \Phi\}$ and \mathcal{Y}^u ;
8: Calculate $P(T(\mathbf{X}_j^u|\Psi)|\Phi)$
9: Update $\Theta = \{\Psi, \Phi\}$ with fixed \mathcal{Y}^u and \mathcal{V} ;
10: $\min_{\Psi} \max_{\Phi} \mathcal{L}^l(\Psi) + \mathcal{L}^u(\mathcal{Y}^u, \Psi, \mathcal{V}) + \beta \mathcal{L}^p(\Psi, \Phi)$
11: **until** converge
12: **return** Model parameter $\Theta = \{\Psi, \Phi\}$;

learning mechanism can be formulated as:

$$\min_{\Theta, \mathcal{V}, \mathcal{Y}^u} \mathcal{L}^l(\Theta) + \mathcal{L}^u(\mathcal{Y}^u, \Theta, \mathcal{V}) + \lambda f(\mathcal{V}), \quad (1)$$

where Θ denotes the parameters of the learning model. $\mathcal{V} = \{\mathbf{V}_1, \mathbf{V}_2, \dots, \mathbf{V}_M\}$ and $\mathcal{Y}^u = \{\mathbf{Y}_1^u, \mathbf{Y}_2^u, \dots, \mathbf{Y}_M^u\}$ indicate the collections of the inferred reliability weight matrices and the pseudo labels (binary and structured) for the unlabeled training images. \mathcal{L}^l and \mathcal{L}^u indicate the loss functions for labeled and unlabeled data, respectively. $f(\mathcal{V})$ is the self-paced regularizer which is usually designed based on the human knowledge in the corresponding task domain. For example, Jiang et al. [9] proposed the linear soft weighting regularizer, logarithmic soft weighting regularizer, and mixture regularizer weighting to build the self-paced re-ranking model for multimedia search. The self-paced regularizer proposed by Zhang et al. [36] consists of a ℓ_1 -norm, a $\ell_{0.5,1}$ -norm, and a Laplacian term for considering the group property in co-saliency detection. Li et al. [37] introduced a ℓ_1 -norm and an adaptive $\ell_{2,1}$ -norm into the self-paced regularizer to simultaneously explore the task complexity and instance complexity for multi-task learning.

Unlike the aforementioned SPL regime, this paper explores a new data-driven strategy, named as APL, to infer the robust learning pace, where the label reliability inference mechanism is implicitly encoded by an additional model, which we call the pace-generator, with undefined but learnable functions. Consequently, APL is equipped with both a task-predictor and a pace-generator, where the task-predictor $T(\cdot)$ generates the task-oriented prediction for the input image and is used to predict the pseudo labels for the unlabeled training images. While the pace-generator $P(\cdot)$ discriminates the reliable and unreliable labels from the obtained pseudo labels dynamically to form a robust learning pace. Then, the whole learning objective function of APL becomes:

$$\begin{cases} \min_{\Psi, \mathcal{V}, \mathcal{Y}^u} \max_{\Phi} \mathcal{L}^l(\Psi) + \mathcal{L}^u(\mathcal{Y}^u, \Psi, \mathcal{V}) + \beta \mathcal{L}^p(\Psi, \Phi), \\ s.t. \mathbf{V}_j = P(T(\mathbf{X}_j^u|\Psi)|\Phi), \forall j = 1, 2, \dots, M, \end{cases} \quad (2)$$

where Ψ and Φ denote the model parameters of the task-predictor and the pace-generator, respectively. By introducing the pace-generator $P(\cdot)$ to infer the reliability of the generated task-oriented labels $T(\mathbf{X}_j^u|\Psi)$ on the unlabeled training images, we have $\mathbf{V}_j = P(T(\mathbf{X}_j^u|\Psi)|\Phi)$. \mathcal{L}^p indicates the objective function for inferring the learning pace under the adversarial-paced learning mechanism, which replaces the self-paced regularizer in Eq.1. β is the free parameters to weigh \mathcal{L}^p . With the underlying relationship between the optimization processes of SPL and GAL, the proposed learning framework can be optimized under a similar pipeline to the conventional SPL methods (see Alg. 1 and Alg. 2) but is able to infer meaningful learning paces through the adversarial-learned pace-generator.

Specifically, we adopt the commonly used cross-entropy loss in \mathcal{L}^l to measure the consistency between the predicted saliency masks and the corresponding human-annotated ground-truth masks:

$$\mathcal{L}^l = - \sum_{i \in \mathcal{D}^l} \Gamma[(\mathbf{1} - \mathbf{Y}_i^l) \log(\mathbf{1} - T(\mathbf{X}_i^l|\Psi)) + \mathbf{Y}_i^l \log T(\mathbf{X}_i^l|\Psi)], \quad (3)$$

where $\Gamma[\cdot]$ indicates the operation to sum all the elements in the input matrix. Different from \mathcal{L}^l , \mathcal{L}^u is defined as a weighted cross-entropy loss, which utilizes the pseudo task-oriented labels generated by the task-predictor as well as the inferred reliability weights as the supervision:

$$\mathcal{L}^u = - \sum_{j \in \mathcal{D}^u} \Gamma[\mathbf{V}_j ((\mathbf{1} - \mathbf{Y}_j^u) \log(\mathbf{1} - T(\mathbf{X}_j^u | \Psi)) + \mathbf{Y}_j^u \log T(\mathbf{X}_j^u | \Psi))], \quad (4)$$

where the element-wise product is used between any two matrixes and it goes the same for all other equations in this paper. To learn a robust learning pace under a data-driven adversarial learning mechanism, we introduce a pace-generator $P(\cdot)$ and define \mathcal{L}^p as:

$$\begin{aligned} \mathcal{L}^p = & \sum_{i \in \mathcal{D}^l} \Gamma[\log P(\mathbf{Y}_i^l | \Phi)] + \sum_{i \in \mathcal{D}^l} \Gamma[\log(\mathbf{1} - P(T(\mathbf{X}_i^l | \Psi) | \Phi))] \\ & + \eta \sum_{j \in \mathcal{D}^u} \Gamma[\log(\mathbf{1} - P(T(\mathbf{X}_j^u | \Psi) | \Phi))]. \end{aligned} \quad (5)$$

By minimizing \mathcal{L}^p , the task-predictor is trained to predict the high-quality task-oriented labels so that the pace-generator would recognize them as the realistic ones, i.e., making $P(T(\mathbf{X}_i^l | \Psi) | \Phi)$ and $P(T(\mathbf{X}_j^u | \Psi) | \Phi)$ close to 1. While by maximizing \mathcal{L}^p , the pace-generator can be trained to discriminate between the generated fake task-oriented labels and the real human annotation, i.e., making $P(\mathbf{Y}_i^l | \Phi)$ close to 1 while $P(T(\mathbf{X}_i^l | \Psi) | \Phi)$ and $P(T(\mathbf{X}_j^u | \Psi) | \Phi)$ close to 0, so that it can acquire the capacity to measure the reliability and truthfulness of the predicted labels.

Optimization. Firstly, we initialize the model parameters $\{\Phi, \Psi\}$ by training the task-predictor and the pace-generator on the labeled training data under a common generative-adversarial learning mechanism:

$$\min_{\Psi} \max_{\Phi} \mathcal{L}^l(\Psi) + \beta (\sum_{i \in \mathcal{D}^l} \Gamma[\log P(\mathbf{Y}_i^l | \Phi)] + \sum_{i \in \mathcal{D}^l} \Gamma[\log(1 - P(T(\mathbf{X}_i^l | \Psi) | \Phi))]). \quad (6)$$

Following the standard GAN training procedure, we adopt a two-stage learning approach to optimize Eq. 6: In the first stage, we learn the parameters of the pace-generator by fixing Ψ . In this case, Φ can be optimized by maximizing $\sum_{i \in \mathcal{D}^l} \Gamma[\log P(\mathbf{Y}_i^l | \Phi)] + \sum_{i \in \mathcal{D}^l} \Gamma[\log(1 - P(T(\mathbf{X}_i^l | \Psi) | \Phi))]$. While in the second stage, we learn the parameters of the task-predictor by fixing Φ . In this case, Ψ can be optimized by minimizing $\mathcal{L}^l(\Psi) + \beta \sum_{i \in \mathcal{D}^l} \Gamma[\log(1 - P(T(\mathbf{X}_i^l | \Psi) | \Phi))]$. The reliability weights in $\{\mathbf{V}_j\}$ are initialized as ones.

After the initialization process, we alternatively infer \mathcal{Y}^u , \mathcal{V} and learn $\{\Phi, \Psi\}$ in each learning iteration. Specifically, we first infer \mathcal{Y}^u and \mathcal{V} based on the network models learned from the previous learning iteration, where \mathcal{Y}^u is obtained by minimizing $\mathcal{L}^u(\mathcal{Y}^u, \Psi, \mathcal{V})$, i.e., forwarding the training images through the learned task-predictor and then binarizing the obtained outputs via the threshold of 0.5, while \mathcal{V} is obtained by following the definition in Eq. 2, i.e., $\mathbf{V}_j = P(T(\mathbf{X}_j^u | \Psi) | \Phi)$. Then, we learn the network parameters $\{\Phi, \Psi\}$ based on the inferred \mathcal{Y}^u and \mathcal{V} :

$$\begin{cases} \min_{\Psi} \max_{\Phi} \mathcal{L}^l(\Psi) + \mathcal{L}^u(\mathcal{Y}^u, \Psi, \mathcal{V}) + \beta \mathcal{L}^p(\Psi, \Phi), \\ s.t. \mathbf{V}_j = P(T(\mathbf{X}_j^u | \Psi) | \Phi), \forall j = 1, 2, \dots, M, \end{cases} \quad (7)$$

Similar to the optimization process of Eq.6, we first learn the parameters of the pace-generator by fixing Ψ :

$$\max_{\Phi} \mathcal{L}^p(\Psi, \Phi), \quad s.t. \mathbf{V}_j = P(T(\mathbf{X}_j^u | \Psi) | \Phi), \quad (8)$$

which encourages the pace-generator to predict the manually annotated labels from \mathcal{D}^l as the realistic ones while predicting the predicted labels from both \mathcal{D}^l and \mathcal{D}^u as the fake ones. In the second stage, we learn the parameters of the task-predictor by fixing Φ :

$$\begin{cases} \min_{\Psi} \mathcal{L}^l(\Psi) + \mathcal{L}^u(\mathcal{Y}^u, \Psi, \mathcal{V}) + \beta \mathcal{L}^p(\Psi, \Phi), \\ s.t. \mathbf{V}_j = P(T(\mathbf{X}_j^u | \Psi) | \Phi), \forall j = 1, 2, \dots, M, \end{cases} \quad (9)$$

which enables the task-predictor to learn informative patterns under the guidance of both the human annotated labels and the confident pseudo labels. The pace loss $\mathcal{L}^p(\Psi, \Phi)$ can also help explore the structure of the predicted labels to regularize the learning of task-predictor. For simplifying the optimization processes of Eq. 8 and Eq. 9, we loose their constraints by converting their constraints to the cross entropy-based loss terms in the learning objective functions.

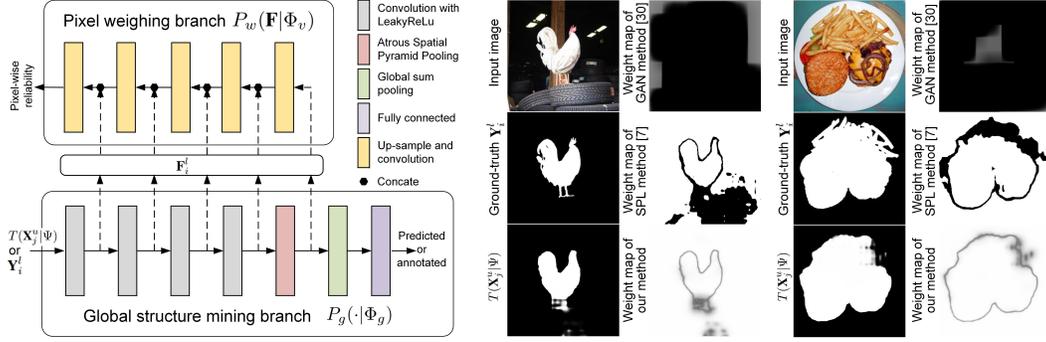


Figure 1: The proposed global structure-guided pixel weighting model and several visual comparison on the weight maps generated by our approach and the conventional GAN or SPL methods. The dashed lines indicate paths without back-propagation. Notice that the displayed weight maps are generated according to the predicted saliency mask $T(\mathbf{X}_j^u|\Psi)$ shown in the bottom-left corner of each set of examples. From the examples, we can observe that the proposed approach can effectively localize the unreliable object boundaries or background regions from the input saliency masks.

Implementation for few-cost salient object detection. We adopt the DeepLab-v2 [38] as the task-predictor $T(\cdot)$ by considering the trade-off between model effectiveness and computational cost. We further alleviate the memory cost by removing the multi-scale fusion module of DeepLab-v2.

When designing the pace-generator $P(\cdot)$, conventional methods, such as [35], might adopt the FCN architecture [39] with an up-sampling layer. They treat all regions of a generated saliency mask as unreliable regions and all regions of a ground-truth saliency mask as reliable regions and train the network parameters in an adversarial learning manner. However, as the generated saliency masks also contain reliable regions that are consistent with the ground-truth, such a learning manner would somehow mislead the learner and may obtain inaccurate weight map \mathbf{V}_j (see Fig. 1). To this end, this paper designs a novel global structure-guided pixel weighting scheme which consists of a global structure mining (GSM) branch $P_g(\cdot)$ and a pixel weighting (PW) branch $P_w(\cdot)$. In the GSM branch, we first use four convolutional layers (with the kernel size of 4×4 , channel number of {64, 128, 256, 512}, and stride of 2) to learn features. Then, we use a global sum pooling layer [40] followed by a fully connected layer to obtain a two-value vector as the prediction of whether the input mask is from model prediction or human annotation. Denote the network parameters in this branch as Φ_g . We train Φ_g in an adversarial learning manner to learn global-structure patterns to infer the reliability of the input mask. To infer the finer pixel-wise reliability of the input mask, the PW branch takes the features extracted by the GSM branch as the inputs and is designed with a set of up-sampling blocks with skip connection to the previous convolutional layers (see Fig. 1). Denote the network parameters in this branch as Φ_v . We train Φ_v in a supervised learning manner by using the ground-truth pixel-wise reliability \mathbf{V}_i^* of the predicted saliency mask on the labeled training images:

$$\mathbf{V}_i^* = 1 - |T(\mathbf{X}_i^l|\Psi) - \mathbf{Y}_i^l|. \quad (10)$$

Then, Eq.5 becomes to:

$$\begin{aligned} \mathcal{L}^P &= \mathcal{L}^{P_g} + \mathcal{L}^{P_w}, \\ \mathcal{L}^{P_g} &= \sum_{i \in \mathcal{D}^l} \log P_g(\mathbf{Y}_i^l|\Phi_g) + \sum_{i \in \mathcal{D}^l} \log(1 - P_g(T(\mathbf{X}_i^l|\Psi)|\Phi_g)) \\ &\quad + \eta \sum_{j \in \mathcal{D}^u} \log(1 - P_g(T(\mathbf{X}_j^u|\Psi)|\Phi_g)), \\ \mathcal{L}^{P_w} &= - \sum_{i \in \mathcal{D}^l} \Gamma[(1 - \mathbf{V}_i^*) \log(1 - P_w(\mathbf{F}_i^l|\Phi_v)) + \mathbf{V}_i^* \log P_w(\mathbf{F}_i^l|\Phi_v)], \end{aligned} \quad (11)$$

where \mathbf{F}_i^l denotes the global structure features extracted by the GSM branch on \mathbf{X}_i^l . Finally, the entire learning objective function becomes to:

$$\begin{cases} \min_{\Psi, \mathcal{V}, \mathcal{Y}^u, \Phi_v} \max_{\Phi_g} \mathcal{L}^l(\Psi) + \mathcal{L}^u(\mathcal{Y}^u, \Psi, \mathcal{V}) + \beta \mathcal{L}^P(\Psi, \Phi_v, \Phi_g), \\ s.t. \mathbf{V}_j = P_w(\mathbf{F}_j^u|\Phi_v) = P(T(\mathbf{X}_j^u|\Psi)|\Phi_v, \Phi_g), \forall j = 1, 2, \dots, M. \end{cases} \quad (12)$$

The optimization process of Eq. 12 still follows the pseudo algorithm shown in Alg. 2. The only difference is that when updating $\Theta = \{\Psi, \Phi_g, \Phi_v\}$ with fixed \mathcal{Y}^u and \mathcal{V} , we optimize the following function for instead:

$$\min_{\Psi, \Phi_v} \max_{\Phi_g} \mathcal{L}^l(\Psi) + \mathcal{L}^u(\mathcal{Y}^u, \Psi, \mathcal{V}) + \beta \mathcal{L}^p(\Psi, \Phi_g, \Phi_v). \quad (13)$$

4 Experiments

We use four widely-used benchmark datasets to implement the experiments, which include PASCAL-S [41], DUT-O [42], SOD [43], and DUTS [28]. Following the previous works [44, 20, 45], we use the training split of the DUT-S dataset for training and test the trained models on the other datasets. Notice that different from the previous works that require the pixel-wise manual annotation on every training images, the approach presented in this work only needs the pixel-wise manual annotation for 1k training images, which is about one-tenth of the whole training images. We use the F-measure and mean absolute error (MAE) to evaluate the experimental results.

We implement the proposed algorithm on the PyTorch framework using a NVIDIA GTX 1080Ti GPU. When training the saliency network, we use the Stochastic Gradient Descent (SGD) optimization method, where the momentum is set to 0.9, and the weight decay is set to 5×10^{-4} . The initial learning rates of the task-predictor and the pace-generator are 2.5×10^{-4} and 10^{-4} , respectively, which are decreased with polynomial decay parameterized by 0.9. For training the pace network, we adopt the Adam optimizer [46] with the learning rate 10^{-4} . The same polynomial decay as the saliency network is also used. We set $\beta = 0.01$ and $\eta = 0.7$ according to a heuristic grid search process. our method uses in total 24.5K iterations and the loss and performance curves are shown in Fig. 2.

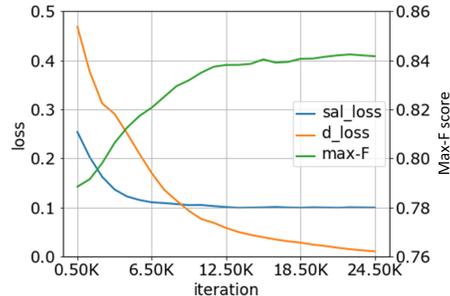


Figure 2: Loss and performance curves on the training split of the DUT-S dataset.

4.1 Comparison to the State-of-the-Art Salient Object Detection Methods

In this section, we compare the proposed approach with 12 state-of-the-art salient object detection methods, which contain both the fully supervised state-of-the-art methods [47, 48, 26, 49, 45, 25, 50, 51] and the unsupervised or semi-/weakly supervised ones [52–54, 28, 34, 55, 56]¹. Notice that all the compared salient object detection models are trained on the same training set but with different amounts or forms of manual annotation. For the fully supervised models, pixel-wise manual annotation of all the training images (about 10k training images) is required. For the unsupervised and weakly supervised models, none pixel-wise manual annotation is required but a certain scope of the image-level annotation is needed. In contrast, the proposed few-cost model uses the pixel-wise manual annotation on only 1k training images. Thus, our approach has much less annotation cost than the conventional fully supervised methods but slightly larger annotation cost than the unsupervised or weakly supervised methods.

The comparison results between our approach and these state-of-the-art salient object detection methods are reported in Table 1 and Fig. 3. From the experimental results, we can observe that our approach outperforms all the state-of-the-art unsupervised or weakly supervised salient object methods with noticeable performance gains. When compared with the existing fully supervised salient object detection methods, our approach can effectively approach the most state-of-the-art method and obtains even better results than some of them. This demonstrates the effectiveness of the proposed approach and implies the rationality of the investigated few-cost salient object detection task in addressing the annotation-hunger issue of the existing salient object detection methods. Some qualitative comparisons of the annotation results are also shown in Fig. 3.

¹We also intend to compare to [6]. However, we are not able to acquire their model or detection results on the datasets used in our comparison.

Table 1: Comparison of the proposed approach with the state-of-the-art salient object detection methods as well as our baseline models on the PASCAL-S, DUT-O, SOD, and DUT-TE datasets.

| Methods | | DUTS-TE | | PASCAL-S | | DUT-O | | SOD | |
|--------------------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | | F_{max} | MAE | F_{max} | MAE | F_{max} | MAE | F_{max} | MAE |
| Fully supervised SOD | F3NET | 0.897 | 0.035 | 0.878 | 0.061 | 0.839 | 0.053 | – | – |
| | EGNet[47] | 0.893 | 0.039 | 0.869 | 0.074 | 0.842 | 0.053 | 0.889 | 0.099 |
| | AFNet[48] | 0.867 | 0.045 | 0.866 | 0.070 | 0.820 | 0.057 | – | – |
| | PoolNet[26] | 0.894 | 0.036 | 0.884 | 0.065 | 0.830 | 0.054 | 0.879 | 0.106 |
| | BASNet[49] | 0.860 | 0.047 | 0.858 | 0.076 | 0.811 | 0.056 | 0.851 | 0.114 |
| | BRN[45] | 0.828 | 0.049 | 0.849 | 0.072 | 0.774 | 0.062 | 0.846 | 0.105 |
| | MLMSNet[25] | 0.854 | 0.048 | 0.858 | 0.074 | 0.793 | 0.064 | 0.862 | 0.108 |
| | PAGE-Net[50] | 0.838 | 0.051 | 0.850 | 0.076 | 0.791 | 0.062 | 0.842 | 0.111 |
| PAGRNet[51] | 0.854 | 0.055 | 0.849 | 0.089 | 0.771 | 0.071 | 0.838 | 0.147 | |
| Un-/semi/weakly supervised SOD | MWS[52] | 0.768 | 0.091 | 0.786 | 0.134 | 0.722 | 0.108 | 0.801 | 0.170 |
| | ASMO[53] | – | – | 0.758 | 0.154 | 0.732 | 0.100 | 0.758 | 0.187 |
| | C2S-NET[54] | 0.807 | 0.062 | 0.842 | 0.082 | 0.758 | 0.072 | – | – |
| | WSS[28] | 0.740 | 0.099 | 0.773 | 0.140 | 0.695 | 0.110 | 0.778 | 0.171 |
| | SGAN[34] | 0.610 | 0.135 | 0.699 | 0.164 | 0.610 | 0.131 | – | – |
| | DeepUSPS[55] | – | – | – | – | 0.736 | 0.063 | – | – |
| SODSA[56] | 0.789 | 0.062 | 0.811 | 0.092 | 0.753 | 0.068 | 0.806 | 0.131 | |
| FC-SOD | Ours | 0.846 | 0.045 | 0.848 | 0.067 | 0.767 | 0.067 | 0.846 | 0.122 |

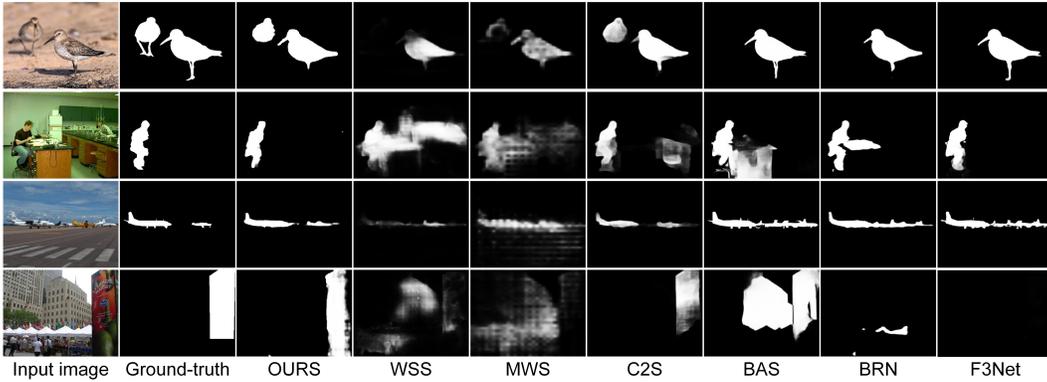


Figure 3: Some examples of the saliency detection results obtained by our approach and other state-of-the-art methods.

4.2 Comparison to the SPL Methods and the Ablation Study Models

In this section, we compare our approach with five self-paced learning schemes [7, 9, 36, 10, 57]. To implement these SPL methods, we replace the \mathcal{L}^p term in our learning object function with their proposed self-paced regularizers and adopt their optimization procedures to train the task-predictor. All other settings are kept the same as the proposed approach. In table 2, we show the comparison results on the DUTS-TE and DUT-O datasets. From the comparison results, we can observe that with the heuristically-designed self-paced regularizers, the conventional SPL methods cannot work well on the investigated task. This indicates the human knowledge embedded in the existing self-paced regularizers is insufficient or inaccurate for few-cost salient object detection.

Besides, we also conduct ablation studies by comparing our approach with five baseline models. The first baseline only uses the \mathcal{L}^l term of Eq. 2, which trains the saliency network by only using the manually annotated training images. The second baseline introduces both the labeled data and unlabeled data in training and considers the pseudo-labels on the unlabeled training images with equal reliability. Based on this baseline, the ‘‘Grab-cut’’ baseline further uses a naive strategy to refine the generated pseudo-labels by using GrabCut [58]. The ‘‘Pixel GAN’’ baseline adopts the conventional pixel GAN [35, 59] to formulate the \mathcal{L}^p term to facilitate the learning process. In this

Table 2: Comparisons to the SPL models.

| Methods | DUTS-TE | | DUT-O | |
|----------------------------|--------------|--------------|--------------|--------------|
| | F_{max} | MAE | F_{max} | MAE |
| ℓ_1 -based [7] | 0.817 | 0.056 | 0.732 | 0.081 |
| LiS-based [9] | 0.819 | 0.055 | 0.732 | 0.081 |
| $\ell_{0.5,1}$ -based [36] | 0.818 | 0.056 | 0.731 | 0.080 |
| $\ell_{2,1}$ -based [10] | 0.820 | 0.055 | 0.733 | 0.079 |
| Fraction-based [57] | 0.818 | 0.055 | 0.729 | 0.080 |
| Ours | 0.846 | 0.045 | 0.767 | 0.067 |

Table 3: Ablation studies.

| | DUTS-TE | | DUT-O | |
|----------------------|--------------|--------------|--------------|--------------|
| | F_{max} | MAE | F_{max} | MAE |
| Only \mathcal{L}^l | 0.817 | 0.059 | 0.719 | 0.085 |
| w/o \mathcal{L}^p | 0.822 | 0.054 | 0.736 | 0.078 |
| GrabCut | 0.730 | 0.120 | 0.616 | 0.129 |
| Pixel GAN | 0.842 | 0.046 | 0.754 | 0.071 |
| w/o V* | 0.840 | 0.046 | 0.752 | 0.068 |
| Ours | 0.846 | 0.045 | 0.767 | 0.067 |

case, our learning objective function degenerates to Eq. 2. Like in Pixel GAN, the “w/o V*” baseline learns our PW branch by constraining all pixels in the predicted saliency masks as the unreliable ones while all pixels in the ground-truth saliency masks as the reliable ones. From the experimental results reported in Table 3, we can observe that simply using the naive GrabCut strategy cannot improve the quality of the generated pseudo-labels. Instead, it would introduce additional noise to the pseudo-labels. Compared to the conventional pixel GAN-based formulation, our approach can better infer the reliability weights for the generated pseudo-labels. It is also interesting to see that the convention SPL methods cannot improve the learning performance of the “w/o \mathcal{L}^p ” baseline. To our best knowledge, this might due to the conventional SPL methods are limited in exploring the structure of the saliency masks and the weight maps obtained by them would hurt the structure of the salient object regions (see examples in Fig. 1).

We have also carried out experiments under different ratios of labeled data with the goal to validate our proposed method’s robustness. The experimental results are reported in Table 4. As can be seen, our approach can learn with different ratios of labeled data. Notably, when only using 1 percent training labels, our approach can still achieve 95.48% performance (in terms of maxF) of the model trained on full training labels.

Table 4: Experiments under different ratios of labeled data.

| | 1% | 5% | 10% | 30% | Full |
|-------|-------|-------|-------|-------|-------|
| max F | 0.824 | 0.840 | 0.846 | 0.846 | 0.863 |
| MAE | 0.054 | 0.049 | 0.045 | 0.044 | 0.044 |

5 Conclusion

This paper studied a problem called few-cost salient object detection. Unlike the conventional salient object detection methods that require large amounts of human annotation, FC-SOD uses only the manual annotation on a few training images together with the annotation-free images. Specifically, we propose an adversarial-paced learning (APL)-based framework to facilitate the few-cost learning scenario. Essentially, APL is derived from the self-paced learning (SPL) regime but it infers the robust learning pace through the data-driven adversarial learning mechanism. For implementing APL for FC-SOD, we further design a global structure-guided pixel weighting scheme to infer the reliability weights for image regions. Comprehensive experiments on widely-used benchmark datasets have demonstrated the effectiveness of the proposed approach. Notably, by using the annotation of only 1k training images, our approach can outperform the existing weakly supervised SOD approaches and perform comparably to the fully supervised SOD models.

Broader Impact

To our best knowledge, this research would provide intelligent visual perception to assistive robotics, which might offer supports in allowing people to live healthier and independent lives for longer. It is believed that advances in automatic saliency detection are net positive for society, despite the potential for misuse. The consequences of the failure of the system would lead to a false understanding of the image content. The task and method do not leverage biases in the data.

References

- [1] J. Han, D. Zhang, G. Cheng, N. Liu, and D. Xu, “Advanced deep-learning techniques for salient and category-specific object detection: a survey,” *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 84–100, 2018.
- [2] D.-P. Fan, W. Wang, M.-M. Cheng, and J. Shen, “Shifting more attention to video salient object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019, pp. 8554–8564.
- [3] D.-P. Fan, T. Li, Z. Lin, G.-P. Ji, D. Zhang, M.-M. Cheng, H. Fu, and J. Shen, “Re-thinking co-salient object detection,” *arXiv preprint arXiv:2007.03380*, 2020.
- [4] Z. Peng, Z. Li, J. Zhang, Y. Li, G.-J. Qi, and J. Tang, “Few-shot image recognition with knowledge transfer,” in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.
- [5] L. Qiao, Y. Shi, J. Li, Y. Wang, T. Huang, and Y. Tian, “Transductive episodic-wise adaptive metric for few-shot learning,” in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.
- [6] Y. Zhou, S. Huo, W. Xiang, C. Hou, and S.-Y. Kung, “Semi-supervised salient object detection using a linear feedback control system model,” *IEEE transactions on cybernetics*, vol. 49, no. 4, pp. 1173–1185, 2019.
- [7] M. P. Kumar, B. Packer, and D. Koller, “Self-paced learning for latent variable models,” in *Advances in Neural Information Processing Systems*, 2010, pp. 1189–1197.
- [8] J. S. Supancic and D. Ramanan, “Self-paced learning for long-term tracking,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2379–2386.
- [9] L. Jiang, D. Meng, T. Mitamura, and A. G. Hauptmann, “Easy samples first: Self-paced reranking for zero-example multimedia search,” in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 547–556.
- [10] L. Jiang, D. Meng, S.-I. Yu, Z. Lan, S. Shan, and A. Hauptmann, “Self-paced learning with diversity,” in *Advances in Neural Information Processing Systems*, 2014, pp. 2078–2086.
- [11] D. Zhang, J. Han, L. Zhao, and D. Meng, “Leveraging prior-knowledge for weakly supervised object detection under a collaborative self-paced curriculum learning framework,” *International Journal of Computer Vision*, vol. 127, no. 4, pp. 363–380, 2019.
- [12] D. Zhang, J. Han, L. Yang, and D. Xu, “Spftn: a joint learning framework for localizing and segmenting objects in weakly labeled videos,” *IEEE transactions on pattern analysis and machine intelligence*, 2018.
- [13] C.-C. Tsai, K.-J. Hsu, Y.-Y. Lin, X. Qian, and Y.-Y. Chuang, “Deep co-saliency detection via stacked autoencoder-enabled fusion and self-trained cnns,” *IEEE Transactions on Multimedia*, vol. 22, no. 4, pp. 1016–1031, 2019.
- [14] K.-J. Hsu, C.-C. Tsai, Y.-Y. Lin, X. Qian, and Y.-Y. Chuang, “Unsupervised cnn-based co-saliency detection with graphical optimization,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 485–501.
- [15] W. Zhang, D. Xu, W. Ouyang, and W. Li, “Self-paced collaborative and adversarial network for unsupervised domain adaptation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.
- [16] K. Ghasedi, X. Wang, C. Deng, and H. Huang, “Balanced self-paced learning for generative adversarial clustering network,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4391–4400.
- [17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
- [18] D. Meng, Q. Zhao, and L. Jiang, “What objective does self-paced learning indeed optimize?” *arXiv preprint arXiv:1511.06049*, 2015.
- [19] N. Liu, J. Han, and 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.
- [20] T. Wang, A. Borji, L. Zhang, P. Zhang, and H. Lu, “A stagewise refinement model for detecting salient objects in images,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4019–4028.

- [21] L. Zhang, J. Zhang, Z. Lin, H. Lu, and Y. He, “Capsal: Leveraging captioning to boost semantics for salient object detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6024–6033.
- [22] W. Wang, J. Shen, M.-M. Cheng, and L. Shao, “An iterative and cooperative top-down and bottom-up inference network for salient object detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5968–5977.
- [23] Z. Wu, L. Su, and 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.
- [24] Y. Liu, Q. Zhang, D. Zhang, and J. Han, “Employing deep part-object relationships for salient object detection,” in *Proceedings of the IEEE international conference on computer vision*, 2019.
- [25] R. Wu, M. Feng, W. Guan, D. Wang, H. Lu, and E. Ding, “A mutual learning method for salient object detection with intertwined multi-supervision,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8150–8159.
- [26] J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, and J. Jiang, “A simple pooling-based design for real-time salient object detection,” *arXiv preprint arXiv:1904.09569*, 2019.
- [27] J. Zhang, T. Zhang, Y. Dai, M. Harandi, and 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.
- [28] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan, “Learning to detect salient objects with image-level supervision,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 136–145.
- [29] D. Zhang, J. Han, Y. Zhang, and D. Xu, “Synthesizing supervision for learning deep saliency network without human annotation,” *IEEE transactions on pattern analysis and machine intelligence*, 2019.
- [30] P. Yan, G. Li, Y. Xie, Z. Li, C. Wang, T. Chen, and L. Lin, “Semi-supervised video salient object detection using pseudo-labels,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7284–7293.
- [31] N. Souly, C. Spampinato, and M. Shah, “Semi supervised semantic segmentation using generative adversarial network,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5688–5696.
- [32] M. Qi, Y. Wang, J. Qin, and A. Li, “Ke-gan: Knowledge embedded generative adversarial networks for semi-supervised scene parsing,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5237–5246.
- [33] J. Dong and T. Lin, “Marginan: Adversarial training in semi-supervised learning,” in *Advances in Neural Information Processing Systems*, 2019, pp. 10 440–10 449.
- [34] C. Wang, S. Dong, X. Zhao, G. Papanastasiou, H. Zhang, and G. Yang, “Saliencygan: Deep learning semi-supervised salient object detection in the fog of iot,” *IEEE Transactions on Industrial Informatics*, 2020.
- [35] W.-C. Hung, Y.-H. Tsai, Y.-T. Liou, Y.-Y. Lin, and M.-H. Yang, “Adversarial learning for semi-supervised semantic segmentation,” *arXiv preprint arXiv:1802.07934*, 2018.
- [36] D. Zhang, D. Meng, and J. Han, “Co-saliency detection via a self-paced multiple-instance learning framework,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 5, pp. 865–878, 2016.
- [37] C. Li, J. Yan, F. Wei, W. Dong, Q. Liu, and H. Zha, “Self-paced multi-task learning,” in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
- [38] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.
- [39] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

- [40] S. Aich and I. Stavness, "Global sum pooling: A generalization trick for object counting with small datasets of large images," in *IEEE Computer Vision and Pattern Recognition (CVPR) Workshop*, 2019.
- [41] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1155–1162.
- [42] C. Yang, L. Zhang, H. Lu, X. Ruan, and 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.
- [43] G. Li and 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.
- [44] N. Liu and 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.
- [45] T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan, and A. Borji, "Detect globally, refine locally: A novel approach to saliency detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3127–3135.
- [46] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [47] J.-X. Zhao, J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "Egnet: Edge guidance network for salient object detection," 2019.
- [48] M. Feng, H. Lu, and E. Ding, "Attentive feedback network for boundary-aware salient object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1623–1632.
- [49] L. Zhang, J. Dai, H. Lu, Y. He, and G. Wang, "A bi-directional message passing model for salient object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1741–1750.
- [50] W. Wang, S. Zhao, J. Shen, S. C. Hoi, and A. Borji, "Salient object detection with pyramid attention and salient edges," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1448–1457.
- [51] X. Zhang, T. Wang, J. Qi, H. Lu, and 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.
- [52] Y. Zeng, Y. Zhuge, H. Lu, L. Zhang, M. Qian, and Y. Yu, "Multi-source weak supervision for saliency detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6074–6083.
- [53] G. Li, Y. Xie, and L. Lin, "Weakly supervised salient object detection using image labels," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [54] X. Li, F. Yang, H. Cheng, W. Liu, and D. Shen, "Contour knowledge transfer for salient object detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 355–370.
- [55] T. Nguyen, M. Dax, C. K. Mummadi, N. Ngo, T. H. P. Nguyen, Z. Lou, and T. Brox, "Deepusps: Deep robust unsupervised saliency prediction via self-supervision," in *Advances in Neural Information Processing Systems*, 2019, pp. 204–214.
- [56] J. Zhang, X. Yu, A. Li, P. Song, B. Liu, and 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. 12 546–12 555.
- [57] Q. Zhao, D. Meng, L. Jiang, Q. Xie, Z. Xu, and A. G. Hauptmann, "Self-paced learning for matrix factorization," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
- [58] C. Rother, V. Kolmogorov, and A. Blake, "' grabcut" interactive foreground extraction using iterated graph cuts," *ACM transactions on graphics (TOG)*, vol. 23, no. 3, pp. 309–314, 2004.
- [59] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.