Multi-source weak supervision for saliency detection

Yu Zeng[1], Yunzhi Zhuge[1], Huchuan Lu[1], Lihe Zhang[1], Mingyang Qian[1], Yizhou Yu[2]
[1] Dalian University of Technology, China
[2] Deepwise AI Lab, China
zengyu@mail.dlut.edu.cn, zgyz@mail.dlut.edu.cn, lhchuan@dlut.edu.cn,
zhanglihe@dlut.edu.cn, mingyangqian25@gmail.com, yizhouy@acm.org

Abstract

The high cost of pixel-level annotations makes it appealing to train saliency detection models with weak supervision. However, a single weak supervision source usually does not contain enough information to train a well-performing model. To this end, we propose a unified framework to train saliency detection models with diverse weak supervision sources. In this paper, we use category labels, captions, and unlabelled data for training, yet other supervision sources can also be plugged into this flexible framework. We design a classification network (CNet) and a caption generation network (PNet), which learn to predict object categories and generate captions, respectively, meanwhile highlight the most important regions for corresponding tasks. An attention transfer loss is designed to transmit supervision signal between networks, such that the network designed to be trained with one supervision source can benefit from another. An attention coherence loss is defined on unlabelled data to encourage the networks to detect generally salient regions instead of task-specific regions. We use CNet and PNet to generate pixel-level pseudo labels to train a saliency prediction network (SNet). During the testing phases, we only need SNet to predict saliency maps. Experiments demonstrate the performance of our method compares favourably against unsupervised and weakly supervised methods and even some supervised methods.

1. Introduction

Saliency detection aims to detect the most informative parts of an image. It can be applied to benefit a wide range of applications [4, 6, 36], and thus has attracted a lot of interest in recent years. Driven by the remarkable success of deep convolutional neural networks (CNNs), a lot of attempts have been made to train CNNs for saliency detection [9, 19, 28, 26]. CNN-based methods usually need a large amount of data with pixel-level annotations for training. Since it is expensive to annotate images with pixel-level ground-truth, attempts have been made to exploit higher-level supervision, *e.g*. image-level supervision, to train CNNs for saliency detection [27].

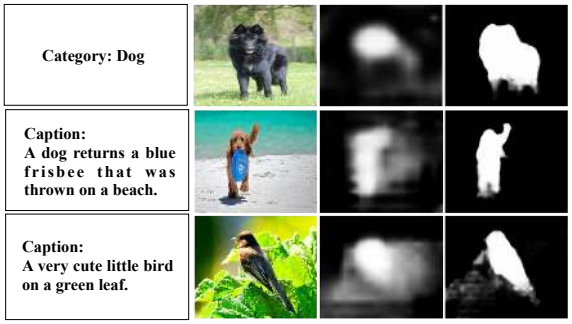

Figure 1. (a) annotations. (b) images. (c) saliency maps of the models trained with single weak supervision source shown in the first column of the corresponding row. (d) saliency maps of the models trained with our proposed multi-source weak supervision framework.

However, it is challenging to train a network to cut the salient objects accurately in weak supervision settings. On the one hand, weak supervision sources are incomplete and noisy. For example, the image-level category label is an efficient weak supervision cue for saliency detection. It indicates the category of the principal objects in it, which are much likely to be the salient foreground. However, category labels are too simple to convey sufficient information. Without knowing the attribute or motion of a salient object, the network trained with category tags might only highlight the most discriminative region instead of the whole object. As shown in the first row of Figure 1, the model trained with category labels only highlights the face of the dog as the face provides enough information to categorize it as a dog. Another weak supervision cue is the image caption. Image captions are a few sentences that describe the main content of an image. Compared with image-level tags, captions provide more comprehensive descriptions of the salient objects. As shown in the second row of Figure 1, for a picture of a

dog, the caption not only tells there is a dog but also says that the dog is returning and is with a frisbee. To generate the correct caption, the network needs to attend the whole dog. Therefore, the network trained with captions is more likely to capture the entire salient objects. However, image captions usually describe not only the salient objects but also the background. This might lead to inaccurate saliency detection results. As shown in the second and the third rows of Figure 1, apart from the salient objects such as the bird and the dog, the captions also mention the background keywords such as the beach and the green leaf. As a result, saliency maps of the networks trained with captions highlight a part of the background.

On the other hand, although it is appealing to integrate multiple weak supervision sources due to their complementarity, there are still plenty of obstacles to it. First, there is a lack of large scale dataset with multiple kinds of annotations, while the existing datasets with different annotations are unmatched for saliency detection task. Second, the models trained by using different annotations are usually required to have different structures. Therefore, it is worth designing a unified framework to combine these models and benefit from multiple sources of annotations.

To this end, we propose a weakly supervised learning framework that integrates multiple weak supervision cues to detect salient objects. Specifically, we use data annotated with image-level tags, image captions, and unlabelled data. Note that other supervision sources also can be plugged into this flexible framework. We design three subnetworks: a multi-label classification network (CNet), a caption generation network (PNet) and a saliency prediction network (SNet). Figure 2 shows the main architecture. CNet is composed of a convolutional feature extractor, an attention module and a fully connected layer. For an input image, the feature extractor produces a feature vector for each region. The attention module generates spatial attention over all regions that control the information flow from each region to the fully connected layer. It has to attend the most important regions to predict the category labels correctly. The spatial attention of all image regions composes a coarse saliency map that highlights all potential category-agnostic object regions. PNet has a similar structure to CNet, with the fully connected layer replaced by an LSTM [8] layer to generate captions. The coarse saliency map generated by its attention module highlights the essential regions for generating correct captions.

To make full use of the annotations, we design an attention transfer loss to transmit supervision signal between networks, such that the network purposed for one supervision source can benefit from another source. When trained with category labels, CNet learns from the annotations, and PNet learns from the coarse saliency maps of CNet with the attention transfer loss. When trained with the images annotated with captions, PNet learns from the annotation, and CNet learns from the coarse saliency maps of PNet. To encourage the networks to detect generally salient regions instead of task-specific regions, we define an attention coherence loss that uses unlabelled data for regularization. The coarse saliency maps of the unlabelled images produced by CNet and PNet are refined according to low-level color similarity, and then coarse saliency maps by CNet and PNet are matched to the refined one.

After CNet and PNet are trained, we use them to generate pseudo labels to train the saliency prediction network (SNet). SNet consists of a feature extractor and several convolution layers. Inspired by [5], we use dilated convolution to enlarge the receptive fields and use parallel dilated convolutional layers with different dilation rates to capture objects and context at multiple scales. When testing, we only need SNet to generate the final saliency maps. As shown in the last column of Figure 1, our proposed multi-source supervision framework can leverage the complementary strengths of diverse supervision sources to generate better saliency maps that evenly highlight the generally salient objects meanwhile suppress the background.

In summary, our main contributions are as follow:

- We propose a novel weak supervision framework to train saliency detection models with diverse supervision sources. As far as we know, this is the first attempt to integrate multiple supervision cues into a unified framework for saliency detection.
- We design three networks for saliency detection that learn from category labels, captions and noisy pseudo labels, respectively.
- We propose an attention transfer loss to transmit supervision signal between networks to let the network designed for one supervision source benefit from another source, and an attention coherence loss to encourage the networks to detect the generally salient regions.

2. Related work

2.1. Salient objects detection

Early research for saliency detection focused on handcrafted features and heuristic priors *e.g*., centre prior [12] and boundary background prior [31]. Recently, driven by the remarkable success of deep convolutional neural networks (CNNs) on various vision tasks, a lot of deep learning based methods have been proposed for saliency detection. Li *et al*. [15] extracted multi-scale features from a deep CNN to represent superpixels and used a classifier network to predict the saliency score of each superpixel. Hou *et al*. [9] proposed a skip-layer structure with deep supervision for saliency detection. Wang *et al*. [29] proposed a

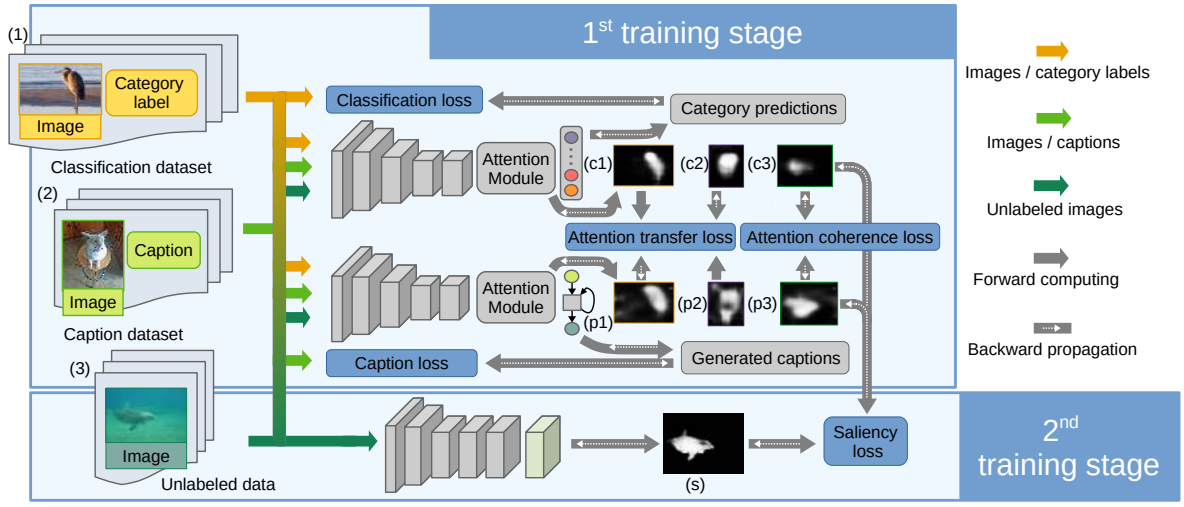

Figure 2. An overview of the proposed multi-source weak supervision framework. (1, 2, 3) images annotated with category labels, caption annotations, and unlabelled images. (c1, c2, c3) saliency maps of images (1, 2, 3) generated by the classification network (CNet). (p1, p2, p3) saliency maps of images (1, 2, 3) generated by the caption generation network (PNet). (s) Final output saliency maps.

global Recurrent Localization Network to exploit contextual information by the weighted response map to localize salient objects more accurately. Although these methods achieved superior performance, they all needed expensive pixel-level annotations for training.

2.2. Weakly supervised learning

To reduce the cost of hand-labelling, weakly supervised learning has attracted increasingly more attention. Pinheiro and Collobert [22] used a segmentation network to predict the pixel-level labels and aggregated them into image-level ones. Then the error between the predictions and the image-level ground truth was backpropagated to update the network. Ahn and Kwak [3] utilized class activation maps (CAM) [34] to train a network to predict semantic affinities within local image areas, which were incorporated with the random walk to revise the CAM and generate the segmentation labels. Wang *et al*. [27] trained a CNN to detect salient objects with image-level supervision. They designed a Foreground Inference Net (FIN) to inference potential foreground regions and proposed a global smooth pooling (GSP) operation to aggregate responses of the inferred foreground objects. Unlike global max pooling (GMP) and global average pooling (GAP) that perform hard selections of latent instances, GSP explicitly computes the weight of each instance and is better suited to pixel-level tasks. However, GSP needs to solve a maximization problem for each input image, which greatly slows down the forward computation of the network. In contrast, our proposed attention module aggregates the features and compute the spatial distribution of foreground objects in one forward pass, bringing much less computation burden. Moreover, all of the above methods rely on a single image-level supervision source, while we integrate complementary supervision cues to train a more robust model.

3. The proposed method

In this section, we elaborate on the proposed multi-source weak supervision for saliency detection. The overall framework is illustrated in Figure 2. The classification network (CNet) predicts category labels meanwhile its attention module generates a coarse saliency map that highlights the regions related to the classification results. The caption generation network (PNet) generates captions and locates the corresponding regions. When training with category labels, the category localization loss is computed for CNet and the attention transfer loss is computed for PNet. When training with captions, the caption localization loss and the attention transfer loss is computed for PNet and CNet respectively. When training with unlabelled data, we compute attention coherence loss using saliency maps of CNet and PNet. After CNet and PNet are trained, we use them to generate pseudo labels to train the saliency prediction network (SNet). The architectures of CNet, PNet as well as the saliency prediction network (SNet) are presented in Section 3.1, 3.2, 3.3. The training strategy is described in Section 3.4.

3.1. Feature extractors

Feature extractors of our networks are designed based on DenseNet-169 [10], which consists of five convolutional blocks for feature extracting and a fully connected linear classifier. Each layer is connected to every other layer

within the same block. Owing to its dense connectivity pattern, DenseNet can achieve comparable classification accuracy with a smaller number of parameters than other architectures. We remove the fully connected classifier and use the convolutional blocks as our feature extractors. To obtain larger feature maps, we remove the downsampling operator from the last few pooling layers. For CNet and PNet, we only do this pruning in the last pooling layer to make the generated feature maps of $\frac{1}{16}$ the input image size. For SNet, we modify the last two pooling layers to obtain feature maps with more detail information and generate better saliency maps. Feature extractor of SNet outputs feature maps of $\frac{1}{8}$ the input image size.

3.2. Attention module

We design an attention module to compute the spatial distribution of foreground objects over the image regions meanwhile aggregate the feature of all regions. Given an input image, the feature extractor generates a feature map denoted as a set of feature vectors $\{\boldsymbol{v}_1, ..., \boldsymbol{v}_K\}$, each of which encodes an image region (*i.e*. a spatial grid location in the last convolutional layer of the feature extractor). K denotes the number of regions and $K = H \times W$ for a feature map of spatial size $H \times W$. We apply a 1×1 convolution followed by a sigmoid function on the feature map to generate a coarse saliency map as follow,

$$s_i = \sigma(\boldsymbol{w}_s^\intercal \boldsymbol{v}_i + b_s), \tag{1}$$

in which σ denotes the sigmoid function. $\boldsymbol{w}_s$ and b_s are the learned parameters. s_i is the saliency score of the i-th region. Saliency scores of all regions constitute a saliency map S.

Given the feature vector $\boldsymbol{v}_i$ and saliency score s_i of each region, we compute its attended feature of each region, denoted as $\boldsymbol{f}_i$ as follow,

$$\boldsymbol{f}_i = s_i \cdot (\boldsymbol{w}_f^\intercal \boldsymbol{v}_i + b_f), \tag{2}$$

in which $\boldsymbol{w}_f$ and b_f are the learned parameters. This can be implemented by another 1×1 convolutional layer of which the output is multiplied with S element-wise. Then we compute a normalized attention weight a_i for each image region as follow,

$$\begin{aligned} a_i &= \boldsymbol{w}_a^\intercal \boldsymbol{f}_i + b_a \\ \boldsymbol{\alpha} &= \text{softmax}(\boldsymbol{a}), \end{aligned} \tag{3}$$

where each element a_i of the vector $\boldsymbol{a}$ is the attention weight of the i-th region. $\boldsymbol{w}_a$ and b_a are the learned parameters. The softmax function is to constrain the sum of the weight of all positions to 1.

Let α_i be the element of $\boldsymbol{\alpha}$; the global attended feature $\boldsymbol{g}$ of the input image is the weighted average of the attended features of all regions as follow,

$$\boldsymbol{g} = \sum_{i=1}^{K} \alpha_i \cdot \boldsymbol{f}_i. \tag{4}$$

This can be regarded as a global pooling operation with adaptive spatial weight. Figure 3 shows the details of the attention module.

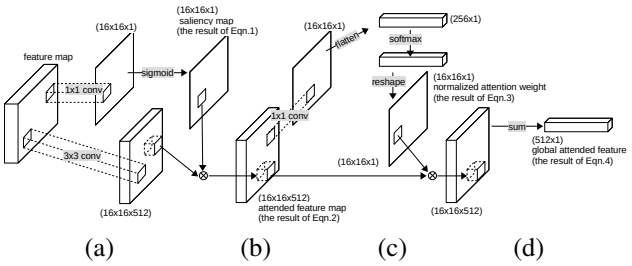

Figure 3. The details of the attention module.

3.3. Network architectures

The classification network (CNet) consists of a previously introduced feature extractor and an attention module, as well as a fully connected layer. Given an input image, the attention module generates its attended global feature and a coarse saliency map from the feature maps provided by the feature extractor. Then the fully connected layer transforms the global attended feature into a C-dimensional vector encoding the probability of each category, in which C is the number of categories.

The architecture of the caption generation network (PNet) is similar to CNet. The main difference between them is that an LSTM layer replaces the fully connected layer of CNet. The LSTM layer takes the global attended feature as input and produces a sequence of M-dimensional vector, in which M is the number of all candidate words. The saliency prediction network (SNet) is composed of a feature extractor; four dilated convolution layers with dilation rates 6, 12, 18, 24 respectively, and a deconvolution layer. The four dilated convolution layers take the feature map as input and predict four saliency maps. Then the four saliency maps are added together and upsampled to the input image size by the deconvolution layer.

3.4. Training with multiple supervision cues

Our training set $\mathcal{D}$ consists of three subsets: the classification dataset, the caption dataset and the unlabelled dataset. The classification dataset is denoted as $\mathcal{D}_c = \{(X^i, \boldsymbol{y}^i\}_{i=1}^{N_c}$, in which $y_j^i \in \{0, 1\}, j = 1, ..., C$ is the one-hot encoding of the categories appearing in image X^i. N_c is the number of samples in $\mathcal{D}_c$. The caption dataset is denoted as $\mathcal{D}_p = \{(X^i, y_{1:T^i}^i\}_{i=1}^{N_p}$, in which $y_{1:T^i}^i$ is a sequence of T^i words $(y_1^i, ..., y_{T^i}^i)$. N_p is the number of samples in $\mathcal{D}_p$.

The unlabelled dataset is denoted as $\mathcal{D}_u = \{X^i\}_{i=1}^{N_u}$, in which N_u is the number of samples.

Given the input image X, CNet predicts the probability of the one-hot label of each category, denoted as $p(y_j|X), j = 1, ..., C, y_j \in \{0, 1\}$, and a saliency map S_c. Each element of S_c, denoted as sc_i, is the saliency score of the i-th region given by Equation 2. PNet outputs the conditional distribution over candidate words at step t of the sequence given the previous words $y_{1:t-1}$, denoted as $p(y_t|y_{1:t-1}, X), y_t = 1, ..., M$. It also generates a saliency map S_p, of which each element is denoted as sp_i.

We define four loss functions to train the networks: a category localization loss L_c, a caption localization loss L_p, an attention transfer loss L_{at} and an attention coherence loss L_{ac}. L_c makes CNet find the most important regions for classification. L_p makes PNet find the most important regions for generating caption. L_{at} transfers supervision signal from the attention map of another network to the current network. L_{ac} encourages two networks supervised by different sources to find the common salient regions. L_c is defined as follow,

$$L_c = -\frac{1}{N_c} \sum_{(X,\boldsymbol{y})\in\mathcal{D}_c} \left[\sum_{j=1}^{C} \log p(y_j|X) + \beta \sum_{s\in S_c} \log(1-s) \right], \tag{5}$$

where the first term is the log-likelihood, and the second term is the regularization that measures the cross-entropy between the saliency map S_c and an all-zero map to prevent the trivial saliency map of having high responses at all locations. β is a hyperparameter set to 0.005. Note that the saliency map S_c and S_p are generated for each input image and thus depend on the input image X. Here and in the following equations, we omit this dependency of symbols for simplification. By minimize Equation 5, CNet learns to predict the categories of the objects present in the input image. Meanwhile, the regularization term limits the amount of information flowing from the image regions to the classifier; therefore the network has to attend on the most important region, *i.e.*, generate a reasonable saliency map, to predict the categories.

The caption localization loss L_p is defined as follow,

$$L_p = -\frac{1}{N_p} \sum_{(X,y_{1:T})\in\mathcal{D}_p} \left[\sum_{t=1}^{T} \log p(y_t|y_{1:t-1}, X) + \beta \sum_{s\in S_p} \log(1-s) \right], \tag{6}$$

where the first term is the log likelihood and the second term is the regularization term as mentioned above. β is set to 0.005. By minimizing Equation 6, PNet learns to generate captions for the input image and find the salient regions corresponding to the caption.

Constrained by is structures, CNet is unable to make direct use of the caption annotations, and PNet is unable to learn from the category annotations directly. To make full use of annotated data, we propose the attention transfer loss to let a network learn from the attention map of another network when its corresponding annotation is not available. Specifically, for an image annotated with category labels, we use the saliency map of CNet to select positive and negative samples (*i.e.* salient regions and background regions) to supervise the saliency map of PNet. For an image annotated with captions, negative and positive samples are selected according to the saliency map of PNet to supervise the saliency map of CNet. Formally, the attention transfer loss is defined as follow,

$$L_{at} = -\frac{1}{N_c} \sum_{(X,\boldsymbol{y})\in\mathcal{D}_c} \left[\sum_{i\in I_c^+} \log sp_i + \sum_{i\in I_c^-} \log(1-sp_i) \right] - \frac{1}{N_p} \sum_{(X,y_{1:T})\in\mathcal{D}_p} \left[\sum_{i\in I_p^+} \log sc_i + \sum_{i\in I_p^-} \log(1-sc_i) \right], \tag{7}$$

where $I_c^+ = \{i|sc_i \geq 0.5\}$ and $I_c^- = \{i|sc_i < 0.5\}$ are the indices of the salient and background regions selected according to the saliency map S_c. $I_p^+ = \{i|sp_i \geq 0.5\}$ and $I_p^- = \{i|sp_i < 0.5\}$ are the indices of the salient and background regions selected according to S_p.

For an input image, CNet and PNet respectively attend to the regions that are most important for predicting categories and generating captions. To make the networks find the generally salient regions, we incorporate low-level color similarity to refine the saliency maps of CNet and PNet and define an attention coherence loss on unlabelled data to match the saliency maps of CNet and PNet to the refined saliency map. Specifically, we segment each unlabelled image into superpixels using SLIC [2] and label the superpixels of which the saliency values are larger than the mean value in both S_c and S_p as salient seed, where the saliency value of a superpixel is defined as the average over its pixels. Then an affinity graph is constructed in which superpixels are nodes. Each superpixel is connected to its two-ring neighbors, and all superpixels on the image boundary are connected. The weight of the edge between the m-th and n-th nodes is defined by the Gaussian of the distance of the Lab color between the corresponding superpixels, *i.e.* $w_{mn} = \exp(-||c_m - c_n||/\sigma^2)$, in which c_m, c_n denote the Lab color of the superpixel m, n, and σ is set to 0.1. Inspired by [31], we rank the color similarity of each superpixel with the salient seed by solving the following problem

of ranking on data manifold [35]:

$$\min_{\boldsymbol{h}} \frac{1}{2}(\sum_{m,n} w_{mn}||\frac{h_m}{\sqrt{d_{mm}}}-\frac{h_n}{\sqrt{d_{nn}}}||^2+\mu\sum_m||h_m-z_m||^2), \tag{8}$$

where $d_{mm} = \sum_n w_{mn}$. μ is set to 0.01. $z_m = 1$ indicates the m-th superpixel is salient seed and $z_m = 0$ otherwise. Let $D = \text{diag}\{d_{mm}\}$, the optimized $\boldsymbol{h}^* = (I-\gamma L)^{-1}\boldsymbol{z}$ is the ranking score of all superpixels, in which $L = D^{-1/2}WD^{-1/2}$ is the normalized Laplacian matrix and $\gamma = 1/(1+\mu)$. We select the pixels of the superpixels whose ranking score is larger than the mean value of $\boldsymbol{h}^*$ as positive samples, denoted as I_u^+, and use other pixels as negative samples, denoted as I_u^-, to supervise the saliency maps of the two networks. The attention coherence loss is defined as follow,

$$\begin{aligned} L_{ac} = -\frac{1}{N_u}\sum_{X\in\mathcal{D}_u}\Bigg[&\sum_{i\in I_u^+}\log sc_i + \log sp_i \\ &+ \sum_{i\in I_u^-}\log(1-sc_i)+\log(1-sp_i)\Bigg]. \end{aligned} \tag{9}$$

The loss function for training the whole system is given by the combination of the above four loss functions:

$$L = L_c + L_p + \lambda L_{at} + \lambda L_{ac}, \tag{10}$$

where λ controls the weight of each term. We use a same weight $\lambda = 0.01$ for L_{at} and L_{ac}.

3.5. Training the saliency prediction network

Having trained CNet and PNet, we use their generated coarse saliency maps to train SNet. The two coarse saliency maps are averaged and resized to the original image size by bilinear interpolation. The averaged map is processed with CRF [14] and then binarized into the pseudo labels. Let Y be the pseudo labels, S the output of SNet. We use the bootstrapping loss [24] to train SNet:

$$\begin{aligned} L_b(S,Y) = &-\sum_i [\delta y_i + (1-\delta)a_i]\log s_i \\ &+ [\delta(1-y_i)+(1-\delta)(1-a_i)]\log(1-s_i), \end{aligned} \tag{11}$$

where y_i, s_i are the elements of Y, S respectively, and $a_i = 1$ if $s_i \geq 0.5$ else $a_i = 0$. δ is set to 0.05. Note that we use CRF only when generating pseudo labels to train SNet. When testing, the saliency maps is predicted in an end-to-end manner without any post-processing.

4. Experiments

4.1. Datasets and evaluation metrics

We evaluate our method on five benchmark datasets: ECSSD [30], PASCAL-S [18], SOD [21], MSRA5K [20] and DUT-OMRON [31]. **ECSSD** contains 1000 natural images with multiple objects of different sizes collected from the Internet. **PASCAL-S** is from the validation set of PASCAL VOC2010 [7] segmentation challenge and contains 850 natural images. **SOD** has 300 images and was designed originally for image segmentation; Jiang *et al*. [11] generated the pixel-wise annotations of salient objects. **MSRA5K** has 5,000 images with a variety of image contents. **DUT-OMRON** contains 5,168 challenging images with one or more salient objects on complex backgrounds.

We use Precision-Recall curve, mean absolute error (MAE) and maximum F-measure (max F_β with β^2 set to 0.3 as suggested in [1]) to quantitatively evaluate the performance of the proposed method and compare with other methods.

4.2. Implementation details

We implement our method using Python with the PyTorch[1] toolbox. Our code will be released for future comparisons[2][3]. In the first training stage, we train CNet and PNet on ImageNet detection dataset for multi-label classification and Microsoft COCO caption dataset as well as about 300,000 images from the ImageNet classification dataset as unlabelled data. In this training stage, we use the Adam optimizer [13] with batch size 36 and learning rate 0.0001. In the second training stage, we use the images of DUTS training set [27] as unlabelled data and use the trained CNet and PNet to generate pseudo ground-truth to train SNet. In this training stage, we use the Adam optimizer with batch size 26 and learning rate 0.0001. All training images are resized to 256×256. During training, we randomly crop and flip the images to avoid overfitting. When testing, the proposed method runs at about 103 fps with 256×256 resolution on our computer with a 3.2GHz CPU, 32GB RAM and two GTX 1080Ti GPUs.

4.3. Ablation studies

In this section, we analyze the contribution of each component including CNet, PNet, the attention transfer loss, the attention coherence loss (applied on unlabelled data), and SNet. The effect of each component in terms of maximum F-measure is shown in Table 1. The visual effect of each component is shown in Figure 4.

Learning from single supervision source. We train CNet and PNet separately to explore the effect of each supervision source. Specifically, CNet is trained with image-level category labels using the category localization loss L_c and PNet is trained with image captions using the caption localization loss L_p. Then we evaluate the performance of each network and the average results of the two networks.

[1] https://github.com/pytorch
[2] https://github.com/zengxianyu/mws
[3] http://ice.dlut.edu.cn/lu/

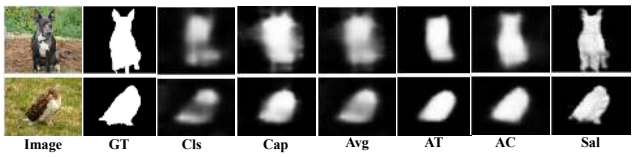

Figure 4. Visual effect of each component. **Image**: input image. **GT**: ground truth. **Cls**: the result of CNet trained with category localization loss L_c. **Cap**: the result of PNet trained with caption localization loss L_p. **Avg**: the average result of **Cls** and **Cap**. **AT**: Jointly training the two networks using the attention transfer loss L_{at}. **AC**: Jointly training the two networks and using unlabelled data for regularization. Loss on the unlabelled data is the attention coherence loss L_{ac}. **Sal**: Training SNet with the pseudo labels generated by CNet and PNet.

As shown in the 1-2 rows of Table 1, both CNet and PNet alone are not able to provide satisfactory results. The average result (the third row of Table 1) is better than each of the two, which demonstrates that the two supervision sources are complementary.

Multi-source supervision with attention transfer loss. Although averaging the results of CNet and PNet can improve the performance, the improvement is minimal. This is because information of training data is not used to the full by training the two networks separately and simply averaging the results. In contrast, by incorporating the attention transfer loss and jointly training the two networks, CNet benefits from the captions and PNet also benefits from the category labels. As a result, jointly training the two networks with attention transfer loss achieves a much better performance (the fourth row of Table 1) than simply averaging the results (the third row of Table 1).

Contribution of the unlabelled data. To verify the contribution of unlabelled data, we train CNet and PNet jointly and using unlabelled data with the attention coherence loss. For labelled data, the loss is the sum of the category (or caption) localization loss and the attention transfer loss. For unlabelled data, we compute the attention coherence loss L_{ac} as in Equation 9. The attention coherence loss encourages the networks to attend on more generally salient objects rather than the task-specific regions. As shown in the fifth row of Table 1, the performance is improved by incorporating unlabelled data and the attention coherence loss.

Effect of the saliency prediction network. After jointly training CNet and PNet with category labels, captions and unlabelled data, we use them to generate pseudo labels to train SNet. The performance of SNet is shown in the last row of Table 1.

4.4. Performance comparison

We compare the performance of our method and 11 stage-of-the-art methods, including five unsupervised methods BSCA [23], MB+ [32], MST [25], MR [31], HS [30], one weakly supervised method WSS [27], and five fully supervised methods DRFI [11], LEGS [26], MCDL [33], MDF [16], DS [17]. The weakly supervised method WSS is trained with category labels of ImageNet detection dataset. The fully supervised methods DRFI, LEGS, MCDL, MDF and DS are trained with pixel-level saliency annotations. Except for DRFI, all the compared supervised methods are based on deep CNNs. We use the saliency maps provided by the authors or obtained by running the code provided by the authors for a fair comparison. The Precision-Recall curves (Figure 5) and the score comparison (Table 2) show that our method outperforms all unsupervised methods with a large margin. As can be seen in Figure 5 and Table 2, the performance of our method is also better than another weakly supervised method WSS. Figure 5 and Table 3 show that our method achieves comparable even better performance against fully supervised methods. As can be seen in Figure 5, our method has a larger recall with the same precision. Table 3 shows that our method outperforms fully supervised methods DRFI and LEGS. Our method also has better performance than MCDL, MDF and DS on most datasets. Visual comparison in Figure 6 also demonstrates the superiority of our method. Compared with unsupervised methods, our methods can detect semantically salient objects that of low contrast to the background, *e.g.* the dog in the first row, and the salient object in the cluttered background *e.g.* the bird in the third row. Compared with another weakly supervised method WSS trained with only object categories, our method can better highlight the non-object salient regions such as water in the fourth and sixth row.

Table 1. Effect of each component in terms of maximum F-measure on ECSSD dataset. **Cls**: Training CNet using category localization loss L_c. **Cap**: Training PNet using caption localization loss L_p. **AT**: Jointly training the two networks with the attention transfer loss L_{at}. **AC**: Jointly training the two networks and using unlabelled data for regularization. Loss on the unlabelled data is the attention coherence loss L_{ac}. **Sal**: Training SNet with the pseudo labels generated by CNet and PNet.

Cls	Cap	AT	AC	Sal	max F_β
√					0.720
	√				0.730
√	√				0.762
√	√	√			0.786
√	√	√	√		0.820
√	√	√	√	√	0.878

5. Conclusion and future work

We propose a unified framework to train saliency detection models with diverse weak supervision sources. We use category labels, captions, and unlabelled data for training. We design A classification network (CNet) and a caption

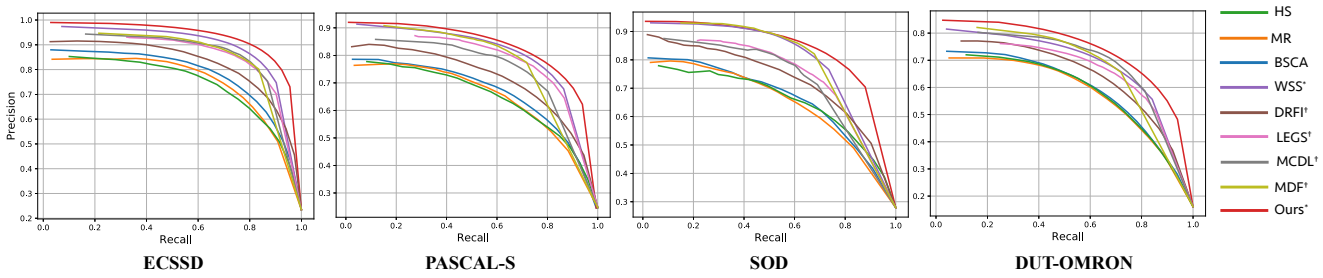

Figure 5. Precision-Recall curves. Our method outperforms unsupervised methods, weakly supervised method (marked with *) and supervised methods (marked with †).

Table 2. Comparison with weakly supervised (marked with * and unsupervised methods in terms of maximum F-measure (the larger the better) and MAE (the smaller the better). The best scores are in bold.

	ECSSD		PASCAL-S		SOD		MSRA5K		DUT-OMRON	
Methods	max F_β	MAE	max F_β	MAE	max F_β	MAE	max F_β	MAE	max F_β	MAE
BSCA	0.758	0.182	0.663	0.223	0.656	0.252	0.829	0.132	0.613	0.196
MB+	0.736	0.193	0.673	0.228	0.658	0.255	0.822	0.133	0.621	0.193
MST	0.724	0.155	0.657	0.194	0.647	0.223	0.809	0.098	0.588	0.161
MR	0.742	0.186	0.650	0.232	0.644	0.261	0.821	0.128	0.608	0.194
HS	0.726	0.227	0.644	0.264	0.647	0.283	0.815	0.162	0.613	0.233
WSS*	0.856	0.104	0.778	0.141	0.780	0.170	0.877	0.076	0.687	0.118
Ours*	**0.878**	**0.096**	**0.790**	**0.134**	**0.799**	**0.167**	**0.890**	**0.071**	**0.718**	**0.114**

Table 3. Comparison with fully supervised methods in terms of maximum F-measure (the larger the better) and MAE (the smaller the better). Weakly supervised method is marked with * . MSRA5K dataset is absent as most supervised methods use it for training.

	ECSSD		PASCAL-S		SOD		DUT-OMRON	
Methods	max F_β	MAE	max F_β	MAE	max F_β	MAE	max F_β	MAE
DRFI	0.785	0.164	0.697	0.207	0.701	0.224	0.651	0.145
LEGS	0.827	0.118	0.761	0.155	0.733	0.196	0.671	0.140
MCDL	0.837	0.101	0.743	0.145	0.730	0.181	0.703	**0.096**
MDF	0.831	0.105	0.768	0.146	0.786	**0.159**	0.693	0.100
DS	**0.882**	0.122	0.763	0.176	0.784	0.190	**0.739**	0.127
Ours*	0.878	**0.096**	**0.790**	0.134	**0.799**	0.167	0.718	0.114

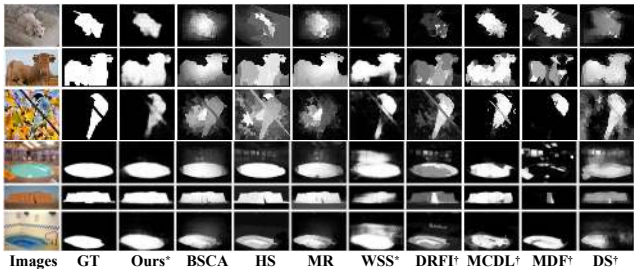

Figure 6. Visual comparison. Weakly and fully supervised methods are marked with * and † respectively.

generation network (PNet), which learn from category labels and captions to generate saliency maps, respectively. An attention transfer loss is designed to transmit supervision signal between networks, such that the network purposed for one supervision source can benefit from another source. An attention coherence loss is defined on unlabelled data to encourage the networks to detect generally salient regions instead of task-specific regions. Final saliency predictions are made by a saliency prediction network (SNet) trained with pseudo labels generated by CNet and PNet. Experiments demonstrate the superiority of the proposed method, of which the performance compares favourably against unsupervised and weakly supervised methods, and is even better than some supervised methods.

The proposed framework is flexible and can be easily extended to integrate more supervision sources. Possible future directions include incorporating more supervision sources such as bounding box supervision, scribble supervision, and noisy saliency maps generated by unsupervised methods. It also can be extended to simultaneously exploit weak supervision sources, unlabelled data, and pixel-level annotations for semi-supervised learning.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (#61725202, #61829102 and #61751212)

References

[1] Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Susstrunk. Frequency-tuned salient region detection. In *IEEE conference on computer vision and pattern recognition*, 2009.

[2] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels. Technical report, 2010.

[3] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *IEEE conference on computer vision and pattern recognition*, 2018.

[4] Shai Avidan and Ariel Shamir. Seam carving for content-aware image resizing. In *ACM transactions on graphics*, volume 26, page 10. ACM, 2007.

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018.

[6] Celine Craye, David Filliat, and Jean-François Goudou. Environment exploration for object-based visual saliency learning. In *IEEE international conference on robotics and automation*, 2016.

[7] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[8] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[9] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip Torr. Deeply supervised salient object detection with short connections. In *IEEE conference on computer vision and pattern recognition*, 2017.

[10] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *IEEE conference on computer vision and pattern recognition*, 2017.

[11] Huaizu Jiang, Jingdong Wang, Zejian Yuan, Yang Wu, Nanning Zheng, and Shipeng Li. Salient object detection: A discriminative regional feature integration approach. In *IEEE conference on computer vision and pattern recognition*, 2013.

[12] Zhuolin Jiang and Larry S Davis. Submodular salient region detection. In *IEEE conference on computer vision and pattern recognition*, 2013.

[13] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[14] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, 2011.

[15] Guanbin Li and Yizhou Yu. Visual saliency based on multiscale deep features. In *IEEE conference on computer vision and pattern recognition*, 2015.

[16] Guanbin Li and Yizhou Yu. Visual saliency detection based on multiscale deep cnn features. *IEEE transactions on image processing*, 25(11):5012–5024, 2016.

[17] Xi Li, Liming Zhao, Lina Wei, Ming-Hsuan Yang, Fei Wu, Yueting Zhuang, Haibin Ling, and Jingdong Wang. Deepsaliency: Multi-task deep neural network model for salient object detection. *IEEE transactions on image processing*, 25(8):3919–3930, 2016.

[18] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *IEEE conference on computer vision and pattern recognition*, 2014.

[19] Nian Liu and Junwei Han. Dhsnet: Deep hierarchical saliency network for salient object detection. In *IEEE conference on computer vision and pattern recognition*, 2016.

[20] Tie Liu, Zejian Yuan, Jian Sun, Jingdong Wang, Nanning Zheng, Xiaoou Tang, and Heung-Yeung Shum. Learning to detect a salient object. *IEEE transactions on pattern analysis and machine intelligence*, 33(2):353–367, 2011.

[21] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *IEEE international conference on computer vision*, 2001.

[22] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *IEEE conference on computer vision and pattern recognition*, 2015.

[23] Yao Qin, Huchuan Lu, Yiqun Xu, and He Wang. Saliency detection via cellular automata. In *IEEE conference on computer vision and pattern recognition*, 2015.

[24] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

[25] Wei-Chih Tu, Shengfeng He, Qingxiong Yang, and Shao-Yi Chien. Real-time salient object detection with a minimum spanning tree. In *IEEE conference on computer vision and pattern recognition*, 2016.

[26] Lijun Wang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Deep networks for saliency detection via local estimation and global search. In *IEEE conference on computer vision and pattern recognition*, 2015.

[27] 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 *IEEE conference on computer vision and pattern recognition*, 2017.

[28] Linzhao Wang, Lijun Wang, Huchuan Lu, Pingping Zhang, and Xiang Ruan. Saliency detection with recurrent fully convolutional networks. In *European conference on computer vision*, 2016.

[29] 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 *IEEE conference on computer vision and pattern recognition*, 2018.

[30] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *IEEE conference on computer vision and pattern recognition*, 2013.

[31] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *IEEE conference on computer vision and pattern recognition*, 2013.

[32] Jianming Zhang, Stan Sclaroff, Zhe Lin, Xiaohui Shen, Brian Price, and Radomir Mech. Minimum barrier salient object detection at 80 fps. In *IEEE international conference on computer vision*, 2015.

[33] Rui Zhao, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Saliency detection by multi-context deep learning. In *IEEE conference on computer vision and pattern recognition*, 2015.

[34] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *IEEE conference on computer vision and pattern recognition*, 2016.

[35] Denny Zhou, Jason Weston, Arthur Gretton, Olivier Bousquet, and Bernhard Schölkopf. Ranking on data manifolds. In *Advances in neural information processing systems*, 2004.

[36] Fabio Zund, Yael Pritch, Alexander Sorkine-Hornung, Stefan Mangold, and Thomas Gross. Content-aware compression using saliency-driven image retargeting. In *IEEE international conference on image processing*, 2013.