Strengthen Learning Tolerance for Weakly Supervised Object Localization

Guangyu Guo[1], Junwei Han[1], Fang Wan[2], Dingwen Zhang[1]*
[1]The Brain and Artificial Intelligence Laboratory, Northwestern Polytechnical University, Xi'an, China
[2]University of Chinese Academy of Sciences, Beijing, China
https://nwpu-brainlab.gitee.io/index_en

Abstract

Weakly supervised object localization (WSOL) aims at learning to localize objects of interest by only using the image-level labels as the supervision. While numerous efforts have been made in this field, recent approaches still suffer from two challenges: one is the part domination issue while the other is the learning robustness issue. Specifically, the former makes the localizer prone to the local discriminative object regions rather than the desired whole object, and the latter makes the localizer over-sensitive to the variations of the input images so that one can hardly obtain localization results robust to the arbitrary visual stimulus. To solve these issues, we propose a novel framework to strengthen the learning tolerance, referred to as SLT-Net, for WSOL. Specifically, we consider two-fold learning tolerance strengthening mechanisms. One is the semantic tolerance strengthening mechanism, which allows the localizer to make mistakes for classifying similar semantics so that it will not concentrate too much on the discriminative local regions. The other is the visual stimuli tolerance strengthening mechanism, which enforces the localizer to be robust to different image transformations so that the prediction quality will not be sensitive to each specific input image. Finally, we implement comprehensive experimental comparisons on two widely-used datasets CUB and ILSVRC2012, which demonstrate the effectiveness of our proposed approach.

1. Introduction

Object detection [22, 18] has made great progress in recent years due to the success of convolutional neural networks (CNNs) [27, 30, 9, 15, 17]. However, the conventional methods would still suffer from the heavy labor costs for providing the bounding box annotations, which makes researchers start paying more attention on the weakly supervised object localization (WSOL) problem [16, 53, 28, 48, 6, 19, 5]. Different from the fully supervised methods, WSOL methods only require the image-level annotations and thus can save a lot of time and labor costs.

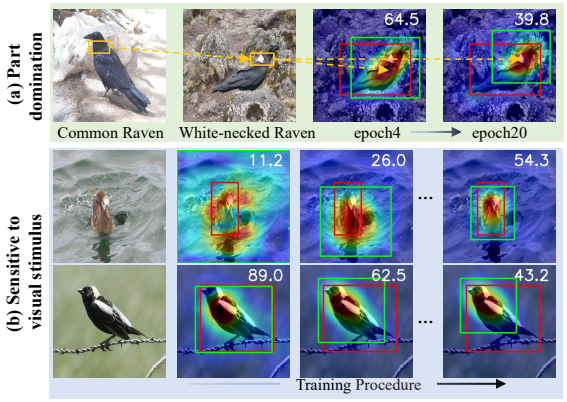


Figure 1. Example of typical WSOL issues. (a) Part domination: focusing on most discriminative parts of the object. (b) Sensitive to visual stimulus: the localization accuracy of different instances show the different convergence trend. The IOU values are shown in white text; the predicted and ground-truth boxes are shown in green and red, respectively.

Previous WSOL methods can be divided into two main categories, *i.e.*, the unified localization-classification framework and the separated localization-classification framework. The former framework predicts the localization map and classification results in the same network which has been exhaustively studied in the prior works [53, 25, 28, 48, 49, 6, 40, 20]. While the latter framework is appeared in two of the most recent works [42, 19]. These methods achieve localization and classification in two separate networks, where off-the-shelf CNN models are directly used as the classifiers. Specifically, in GC-Net [19], the authors first generate a geometry constrained mask to split foreground and background. Then, they optimize the obtained mask under the guidance of a trained classifier. Different form GC-Net, PSOL [42] directly uses an co-localization methods DDT [53] to generate pseudo bounding boxes for the localization branch. Since the unified localization-classification framework has to change the network architecture to obtain better localization performance, the sepa-

*Corresponding author.

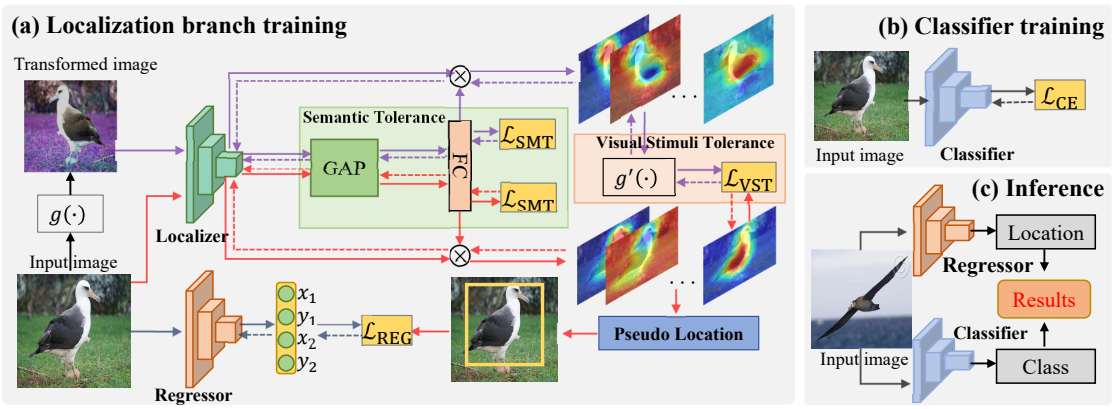


Figure 2. SLT-Net consists of the localizer, regressor, and classifier. (a) When train the localization branch, the input image and transformed image are fed into the localizer to get two class activate maps (CAM). Then inverse transformations are applied to the CAM of the transformed image, and the inverse transformed CAM is matched to the original CAM. The regressor is supervised by the predicted location of the localizer. (b) The classifier is trained independently. (c) In the inference process, we get the classification and localization results from the classifier and regressor, respectively.

rated localization-classification framework will have higher classification accuracy. In this paper, we adopt to use the separated localization-classification framework.

For both of the aforementioned WSOL frameworks, one typical problem is the part domination effect of CNNs: predictions tend to be dominated by the most discriminative parts of the object. As shown in Figure 1a, since the *Common Raven* category and *White-necked Raven* category has little difference except the color of the neck regions, the class activate maps extracted from these images would only focus on the neck of the bird, which will lead to an incorrect prediction of the object location. We believe that the reason for this phenomenon is the lack of tolerance to the semantic mistakes. In other words, network models used for localization should not be punished too much for predicting a similar but wrong semantic class. Another problem in WSOL is the learning robustness issue. While only the image-level supervision is available, the model can hardly extract the equivariant patterns during the learning process. This makes the models sensitive to the variations of the input visual stimulus, such as different hues, contrast, texture, spatial location, *etc*. Consequently, the convergence trends of different instances' localization accuracy become quite different. As shown in Figure 1b, although the two images belong to the same category *Bobolink*, the localization accuracy of the top image increases in the whole training process. On the contrary, the localization result of the bottom image is accurate in the first training epoch and then drifts to inaccurate regions at the terminate iteration. This phenomenon makes it hard to obtain a WSOL model that can achieve accurate performance for the arbitrary input images. Maybe it is hard to eliminate the variations of the input images, but we can alleviate this problem by enhancing the model's tolerance to such variations on our own initiative.

Based on the above considerations, in this paper, we propose a novel weakly supervised object localization framework to strengthen the learning tolerance to 1) the mistakes made in semantic classifying and 2) different image transformations. We name the proposed methods the SLT-Net. As shown in Figure 2, SLT-Net is a separate localization-classification framework that contains three sub-networks: a localizer, a regressor, and a classifier. The localizer and regressor compose the localization branch that provides the location predictions, while the classifier takes responsibility for predicting the class. To make the localizer tolerate the semantic classification mistakes, we design a class-tolerant classification module and the corresponding class-tolerant activation mapping technique to generate the location map. Specifically, for each input image, we divide all candidate categories into two groups: one contains the correct category together with its similar categories, while the other contains the categories that of less similarity to the correct category. For the categories from the former group, we reduce their training losses to make the model alleviate the part dominant effect. For the categories from the later group, we use the normal classification training strategy to improve the ability of the localizer to distinguish the foreground and background. Moreover, to enhance the tolerance for different input visual stimulus, we first apply image transformations to actively improve the variability of the training images and then enable the model to learn the desired equivariant patterns by minimizing the difference between the class activates maps of the transformed image and the original image.

To sum up, this work mainly contains the following fourfold contributions:

- We propose a novel separated localization classification method SLT-Net for weakly supervised object localization. SLT-Net improves localization performance by strengthening the learning tolerance to se-

mantic mistakes and data distribution diversity.

- We propose a class-tolerance classification module to strengthen the tolerance of semantic classification mistakes, which can mitigate the part domination problem by reducing the punishment of error classification among similar categories.
- We strengthen the tolerance to image diversity by matching the visual response map of the transformed image to that of the original image.
- Experiments on the fine-grained dataset CUB and large-scale dataset ILSVRC2012 demonstrate the effectiveness of the proposed method.

2. Related Works

Unified localization-classification weakly supervised object localization. WSOL aims to locate the object when only the image-level annotation is available. Based on the success of CNNs in classification, Simonyan et al. propose a visualization technique that can compute a class saliency map for WSOL [26]. Class activation map (CAM) [53] introduced the global average pooling (GAP) layer behind the final feature of a CNN to generate class-specific localization maps and find discriminative regions. Grad-cam [25] and Grad-cam++ [3] utilize the gradients flow to produce the localization map and do not need the GAP layer. Afterward, many methods are proposed to improve the localization performance by erasing the discriminative regions of the image or feature, forcing the networks to capture a more part of the object region rather than its most discriminative region [28, 48, 6, 20]. SPG [49] utilizes the discriminative regions from the latter layers to train the earlier layers. CutMix [41] explores a data augmentation technique to simultaneously improve classification and localization performance by mixing objects of different categories. Bae et al. [1] proposed several techniques to resolve the bias of global average pooling and instability of thresholding reference in CAM. I2C [50] take inter-class relation into consideration and achieves a good performance.

Separated Localization-Classification Weakly Supervised Object Localization. Unlike methods that achieve the classification and localization by the same network, PSOL [42] achieves the classification and localization task by two separate networks, and the localization network is trained by pseudo bounding boxes that are generated by a class-agnostic co-localization method DDT [39]. Moreover, GC-Net [19] also utilizes two separate networks in the inference process. GC-Net first trains a classifier, then fixes its parameters and train the detection network. In this paper, we propose a separate localization-classification method SLT-Net for weakly supervised object localization. We enhance the semantic tolerance of the localizer by exploring a new way to utilize the classification label. Moreover, we propose a novel training strategy to make the model insensitive to the inconsistency of images.

Weakly supervised object detection. Weakly supervised object detection (WSOD) aims to train a detector by weaker supervision such as image-level labels [2, 36, 45, 46]. Most methods handle WSOD as a multiple instance learning problem, in which object proposals must be provided and the model is trained to select the the most confident proposal [2, 33, 32, 37, 35, 23]. Bilen *et al*. proposed an end-to-end architecture for WSOD [2]. Then many methods were proposed to improve the performance by multi-stage refinement [33, 32, 4], better optimization strategy [37, 37], curriculum learning [47, 43, 45] *etc*. Some researchers also jointly training weakly supervised object detection and weakly supervised segmentation by multi-task learning [11, 12, 44]. Recently, there are some work jointly using audio-visual cues for weakly-supervised object localization [52, 8, 51, 10].

3. Methods

Given a series of images and only image-level labels from C categories, the goal of WSOL is to train a model that can precisely locate and classify the objects in the images. In this section, we first present the overview of the proposed SLT-Net, then we give a brief review of the class activation mapping (CAM) [53], which is used as the baseline in this paper. Finally, we give a detailed description of each module of SLT-Net.

3.1. Overview

As shown in Figure 2, the proposed SLT-Net consists of three networks: the localizer that is trained using the image-level annotations and aims to predict the location implicitly, the regressor that is trained by the prediction of the localizer, and the classifier that aims to predict the classification results. The loss of the proposed SLT-Net is defined as

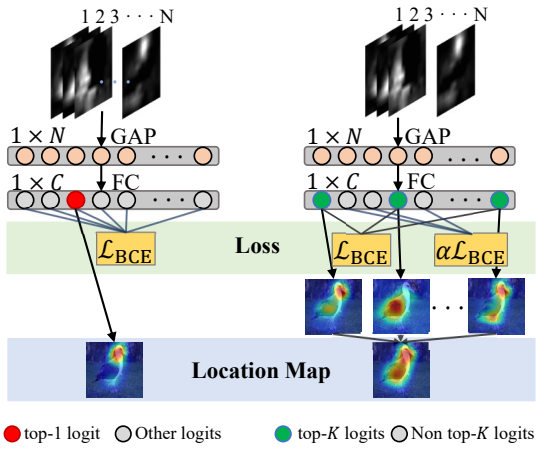
$$\mathcal{L}_{\text{SLT-Net}} = \mathcal{L}_{\text{LOC}} + \mathcal{L}_{\text{REG}} + \mathcal{L}_{\text{CLS}}. \tag{1}$$

In the training process of the localization branch, the localizer and the regressor are trained synchronously. As shown in Figure 2a, the localizer consists of a semantic tolerance module (SMT) and visual stimuli tolerance module (VST). In every single iteration, its input is a training image and its transformed image, and the loss function is:

$$\mathcal{L}_{\text{LOC}} = \mathcal{L}_{\text{SMT}} + \beta\mathcal{L}_{\text{VST}} \tag{2}$$

where $\mathcal{L}_{\text{SMT}}$ is the semantic mistakes tolerance loss and $\mathcal{L}_{\text{VST}}$ is the visual stimulus tolerance loss, which will be described in Section 3.3 and 3.4 respectively. β is a trade-off hyperparapeter.

Then we use the prediction map of the localizer to generate the bounding box $\mathbf{p} = (\frac{x_1}{m}, \frac{y_1}{m}, \frac{x_2}{m}, \frac{y_2}{m})$, where (x_1, y_1) are the top-left coordinate of the bounding box, (x_2, y_2) are the bottom-right coordinate of the bounding box, and m is the downsampling factor. This prediction map of the localizer is used as the pseudo label to train the regressor. Different from the localizer, the regressor predicts the location



(a) CAM (b) top-K ACAM

Figure 3. Architecture of the CAM (a) and the proposed class-tolerance classification module (b).

in an explicit way. In this paper, we build the regressor by adding two fully connected layers and corresponding ReLU layers behind the backbone network, we utilize the Smooth L1 loss [13] as the regression loss $\mathcal{L}_{\text{REG}}$:

$$\mathcal{L}_{\text{REG}}(\mathbf{p}, \hat{\mathbf{p}}) = \sum_{i=1}^{4} \text{smooth}_{L_1}(\mathbf{p}_i - \hat{\mathbf{p}}_i) \tag{3}$$

in which

$$\text{smooth}_{L_1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise} \end{cases} \tag{4}$$

The classifier is trained independently from the localizer and regressor. Given the predicted logits $\mathbf{x} = [x_1, x_2, ..., x_C]$, and ground truth $\mathbf{y} = [y_1, y_2, ..., y_C], y_i \in \{0, 1\}$, the loss of the classifier is the cross entropy loss:

$$\mathcal{L}_{\text{CLS}}(\mathbf{x}, \mathbf{y}) = \mathcal{L}_{\text{CE}}(\mathbf{x}, \mathbf{y}) = -\sum_{i=1}^{C} y_i \log x_i \tag{5}$$

In the inference process, we get classification predictions from the classifier and the locations from the regressor. In this paper, we mainly explore how to improve the performance of the localization branch, which is presented in the following sections.

3.2. Review of Class Activation Map (CAM)

Even though many methods have been proposed to solve the WSOL problem in recent years, most of them are still based on CAM [53]. Figure 3a shows the framework of CAM. Suppose the feature map of spatial size $W \times H$ and N channels as $\mathbf{F} \in \mathbb{R}^{W \times H \times N}$. CAM first feeds the final feature map $\mathbf{F}$ into the global average pooling (GAP) layer. Let the n-th channel of $\mathbf{F}$ be $\mathbf{F}_n$, it is pooled into a vector by the GAP layer:

$$\mathbf{v}_n = \frac{1}{W \times H} \sum_{(w,h)} \mathbf{F}_n(w, h) \tag{6}$$

where $\mathbf{v}_n \in \mathbb{R}^{1 \times 1}$ is the pooled feature vector of $\mathbf{F}_n$. $\mathbf{F}_n(w, h)$ denotes the element of position (h, w).

Then the pooled feature vector $\mathbf{v} \in \mathbb{R}^{1 \times N}$ is feed to a fully connected layer. Let the parameters of the fully connected layer as $\mathbf{W} \in \mathbb{R}^{N \times C}$, CAM of class c is:

$$\mathbf{M}_c = \sum_{n=1}^{N} w_{n,c} \cdot \mathbf{F}_n \tag{7}$$

where $w_{n,c}$ is the element of $\mathbf{W}$ and $\mathbf{M}_c \in \mathbb{R}^{W \times H}$. The total class activate maps of the input image is $\mathbf{M} = [\mathbf{M}_1, \mathbf{M}_2, ..., \mathbf{M}_C]$. Then CAM segments the regions with the top 20% values and chose the bounding box that covers the largest connected component as the predicted localization result.

CAM trains the model as a classifier, its performance in terms of GT-Known LOC is pretty low, but the peak localization performance of CAM in the training process is pretty high. In this paper, we adopt CAM as our base localizer and propose learning tolerance mechanisms to tackle the part domination and learning robust issue.

3.3. Learning Tolerance to Semantic Mistakes

The conventional WSOL works always regard the model as a classifier and use the cross entropy loss as the the classification loss. However, training a WSOL model as a classifier will make the model has little tolerance for the semantic classification mistakes and lead to the part domination problem. In this paper, as it is unnecessary to consider the classification performance of the localizer under a separated localization-classification framework, we propose a novel classification module that aims to strengthen the tolerance to the classification mistakes, which guide the localization branch to obtain the best localization results.

Figure 3b shows the architecture of the semantic mistakes tolerance module (SMT). We think the main reason why traditional classification will lead to part domination problem is that the network has to accurately distinguish one category from other categories that have a similar appearance to it, so in the proposed semantic mistakes tolerance module, we reduce the loss by a factor of $\alpha, 0 \leq \alpha \leq 1$ when the top-K predicted scores (top-$K(\mathbf{x})$) contain the correct ground truth, which would make the loss have less effect on the weights of the network in the backpropagation. Suppose the k-th category is the correct label, *i.e.*, $y_k = 1$, the loss of the semantic mistakes tolerance module is :

$$\mathcal{L}_{\text{SMT}}(\mathbf{x}, \mathbf{y}) = \begin{cases} \alpha \cdot \mathcal{L}_{\text{CE}}(\mathbf{x}, \mathbf{y}), & \text{if } x_k \in \text{top-}K(\mathbf{x}) \\ \mathcal{L}_{\text{CE}}(\mathbf{x}, \mathbf{y}), & \text{otherwise} \end{cases} \tag{8}$$

value K is used to control how many categories are considered highly relevant to the real category, and α is used to control the reduction of punishment. This loss will become the cross entropy loss when $K = 1$ and $\alpha = 1$.

This kind of design would bring two benefits. Firstly, compared to previous WSOL methods that regard the model as a classifier, the semantic mistakes tolerance module would significantly relieve the part domination problem. The localizer does not need to distinguish one category from

Table 1. The quantitative results on CUB dataset when VGG16 is used as backbone. *denotes results not from the original paper and the reference behind it is the data sources. The results of the first and second are shown in red and green, respectively.

Method	Top-1 CLS	Top-1 LOC	GT-known LOC
VGG16-CAM [53]	**76.6**	44.2	58.0
VGG16-ACoL [48]	71.9	45.9	59.3
VGG16-SPG [49]	75.5	48.9	-
VGG16-HaS-32* [1]	66.1	49.5	71.6
VGG16-ADL [6]	65.3	52.4	**75.4**
VGG16-DANet [40]	75.4	52.5	-
VGG16-MEIL [20]	74.8	57.5	73.8
VGG16-PSOL [42]	-	**66.3**	-
VGG16-GC-Net [19]	**76.8**	63.2	-
VGG16-SLT-Net	**76.6**	**67.8**	**87.6**

Table 2. The quantitative results on CUB dataset when InceptionV3 is used as backbone. * denotes results not from the original paper and the reference behind it is the data sources. The results of the first and second are shown in red and green, respectively.

Method	Top-1 CLS	Top-1 LOC	GT-known LOC
GoogLeNet-CAM [53]	73.8	41.1	-
GoogLeNet-HaS-32* [1]	75.4	47.4	61.1
GoogLeNet-ADL* [1]	73.4	51.3	66.8
GoogLeNet-SPG [49]	-	46.6	-
GoogLeNet-DANet [40]	71.2	49.5	-
GoogLeNet-GC-Net [19]	**76.8**	58.6	-
InceptionV3-ADL [6]	74.6	53.0	-
InceptionV3-PSOL [42]	-	**65.5**	-
InceptionV3-I2C [50]	-	56.0	**72.6**
InceptionV3-SLT-Net	**76.4**	**66.1**	**86.5**

its $K-1$ most similar categories precisely. Secondly, compared to PSOL [42] that uses an unsupervised method as the localizer, the proposed semantic mistakes tolerance module can help the localizer to maintain certain discrimination ability to reduce the false response on the background, as we use the classification loss when the correct category is in the $K+1$ to C-th predicted scores.

As the objective function will also makes CAMs of Top-K predicted categories contain the response to the object, so in the semantic mistakes tolerance module, the final localization map is generated by averaging the CAMs of the top-K predicted categories (top-K ACAM). Let $\mathbf{t} = [t_1, t_2, ..., t_K]$ be the top-K categories in the predicted scores, the localization map $\mathbf{A}$ is generated by:

$$\mathbf{A} = \frac{1}{K}\sum_{j\in\mathbf{t}}\mathbf{M}_j \tag{9}$$

where $\mathbf{A} \in \mathbb{R}^{W\times H}$.

3.4. Learning Tolerance to Visual Stimulus

As shown in Figure 1b, the diversity of visual stimulus makes the WSOL model can only obtain accurate location on the part of images. The visual stimulus of images is composed by virous factors and it is impossible to eliminate its diversity, but we can enhance WSOL models' tolerance to it. In this paper, we design a training strategy to reduce the influence of the visual stimulus on the localizer. As shown in Figure 2a, we first transform the original image $\mathbf{I}$ by several transformations:

$$\mathbf{I}^{\mathrm{t}} = g(\mathbf{I}) \tag{10}$$

where $\mathbf{I}^{\mathrm{t}}$ is the transformed image, $g(\cdot)$ denotes the visual transformations, we use several transformations like color jitter (includes brightness, contrast, saturation, and hue), horizontally flip, scale. Then we fed the original image and transformed the image into the localizer, and get the two class activate maps:

$$\mathbf{M}^{\mathrm{o}} = f_{loc}(\mathbf{W}_{loc}, \mathbf{I}) \qquad \mathbf{M}^{\mathrm{t}} = f_{loc}(\mathbf{W}_{loc}, \mathbf{I}^{\mathrm{t}}) \tag{11}$$

where $f_{loc}(\cdot)$ and $\mathbf{W}_{loc}$ denote the inference operation and weights of the localizer, respectively. $\mathbf{M}^{\mathrm{o}} \in \mathbb{R}^{W\times H\times C}$ is the class acitvate map of the original image, $\mathbf{M}^{\mathrm{t}} \in \mathbb{R}^{W^{'}\times H^{'}\times C}$ is the class acitvate map of the transformed image. Then we utilize inverse transformation to $\mathbf{M}^{\mathrm{t}}$:

$$\mathbf{M}^{\mathrm{it}} = g^{'}(\mathbf{M}^{\mathrm{t}}) \tag{12}$$

where $g^{'}$ denote the reverse transform operation, $\mathbf{M}^{\mathrm{it}} \in \mathbb{R}^{W\times H\times C}$. It is notable that the inverse transformation is only utilized for spatial transformations like horizontally flip, scale.

To makes the predicted location of the localizer insensitive to the diversity of visual stimulus like color or spatial jitter (*i.e.*, transformations we used), we think that the CAM of $\mathbf{I}$ should as similar as possible to the inverse transformed CAM of $\mathbf{I}^{\mathrm{t}}$, so we use mean squared error loss (l_2 loss) as the visual stimulus tolerance loss:

$$\mathcal{L}_{\mathrm{VST}} = \frac{1}{W\times H}\| \mathbf{M}^{\mathrm{o}} - \mathbf{M}^{\mathrm{it}} \|_2^2 \tag{13}$$

By such a design, the localizer will insensitive to the transformations we have used to generate $\mathbf{I}^{\mathrm{t}}$, so the tolerance of the localizer to the visual stimulus will increase.

Similar strategy has been used in multi-label image classification [14] and weakly supervised semantic segmantation [38], but in this paper it is designed for a different purpose as it aims at strengthening the learning tolerance to visual stimulus.

4. Experiments

4.1. Experimental Setup

Datasets. We evaluate our method on two wildly used WSOL benchmarks, *i.e.*, CUB (Caltech-UCSD Birds-200-2011) [34] and ILSVRC2012 [7, 24]. CUB is a fine-grained dataset containing 200 categories of birds, and consists of 5,994 training images and 5,794 testing images. ILSVRC2012 is a large-scale classification dataset with 1000 categories. We use the subset5[1] that contains 1.2 million training images and 50 thousand validation images.

[1]This subset has not been changed since 2012

Table 3. The quantitative results on ILSVRC2012 dataset when VGG16 is used as backbone. The results of the first and second are shown in red and green, respectively.

Method	Top-1 CLS	Top-1 LOC	Top-5 LOC	GT-known LOC
Backprop [26]	-	38.9	48.5	-
VGG16-CAM [53]	68.8	42.8	54.9	59.0
VGG16-ACoL [48]	67.5	45.8	59.4	63.0
VGG16-ADL [6]	69.5	44.9	-	-
VGG16-MEIL [20]	**70.3**	46.8	-	-
VGG16-PSOL [42]	-	**50.9**	**60.9**	**64.0**
VGG16-I2C [50]	69.4	47.4	58.5	63.9
VGG16-SLT-Net	**72.4**	**51.2**	**62.4**	**67.2**

Metrics. Following previous works [42, 50], the classification performance is measured by Top-1/Top-5 accuracy (Top-1/Top-5 CLS), the localization performance is measured by Top-1/Top-5 accuracy (Top-1/Top-5 LOC) and with known ground-truth class (GT-known LOC). Specifically, Top-1/Top-5 CLS is correct if the Top-1/Top-5 predict categories contain the correct label. GT-known LOC is correct when the intersection over union (IoU) between the ground-truth and the prediction is larger than 0.5 and does not consider whether the predicted category is correct. Top-1/Top-5 LOC is correct when Top-1/Top-5 CLS and GT-Known LOC are both correct. Since the classification performance is not considered when training the localizer, we only utilize the GT-known LOC metric to evaluate the performance of localizer.

Implementation details. We use VGG [27] and InceptionV3 [31] as the backbone of the classifier and regressor. Of note, we used the same backbone for the classifier and regressor. For the localizer, according to previous unified classification-localization methods [53, 40, 49], we replace the last pooling layer and two fully connected layers of VGG16 with a GAP layer and remove the last three convolution blocks of InceptionV3.

While training the localizer and regressor, we use mini-batch stochastic gradient descent (SGD) [29] as the optimizer, the momentum, and the weight decay is set as 0.9 and 0.0005, respectively, the learning rate is 0.0002, and batch size is 16. We train 2 epochs on ILSVRC2012 and 4 epochs on CUB. On ILSVRC2012 dataset, we set $\alpha = 0.2$, $\beta = 10$, $K = 100$. For experiments on CUB, set $\alpha = 0.4$, $K = 30$, and keep other hyperparameters the same as ILSVRC2012. In the training process of localizer, regressor and classifier, following [19] and [42], if the backbone is VGG16, we resize the input image to size 256×256 and then crop it to 224×224. If InceptionV3 is used as the backbone, we resize the input image to size 320×320 and crop it to 299×299. For testing models, following previous works [53, 6, 19], we use ten crop augmentations to get final classification results and use single image input for all our localization results.

Table 4. The quantitative results on ILSVRC2012 dataset when InceptionV3 is used as backbone. The results of the first and second are shown in red and green, respectively.

Method	Top-1 CLS	Top-1 LOC	Top-5 LOC	GT-known LOC
GoogLeNet-Backprop [26]	-	38.7	49.5	-
GoogLeNet-GMP [53]	64.4	42.2	54.7	-
GoogLeNet-CAM [53]	65.0	43.6	57.0	-
GoogLeNet-HaS-32 [28]	70.7	45.2	-	60.3
GoogLeNet-ACoL [48]	71.0	46.7	57.4	-
InceptionV3-CAM [53]	73.3	46.3	58.2	62.7
InceptionV3-SPG [49]	-	48.6	60.0	-
InceptionV3-DANet [40]	72.5	47.5	58.3	-
InceptionV3-ADL [6]	72.8	48.7	-	-
InceptionV3-PSOL [42]	-	**54.8**	63.3	65.2
InceptionV3-MEIL [20]	73.3	49.5	-	-
InceptionV3-GC-Net [19]	**77.4**	49.1	58.1	-
InceptionV3-I2C [50]	73.3	53.1	**64.1**	**68.5**
InceptionV3-SLT-Net	**78.1**	**55.7**	**65.4**	**67.6**

We implement our method on python and Pytorch [21] [2].

4.2. Ablation studies

Ablation studies about the localizer. We perform ablation studies to analyze the localizer on both CUB and ILSVRC2012 dataset and report the results in Table 5, all the results are measured by GT-Know LOC metric, and the backbone is CAM [53]. As shown in Table 5, VGG16-CAM achieves 59.0% and 58.0% on CUB and ILSVRC2012 dataset, respectively. However, CAM considers both classification and localization performance. Still, the localizer in our approach only takes localization performance into consideration, so we change the optimization strategy to achieve the best localization performance. Specifically, we train the localizer with a lower learning rate (from 0.001 to 0.0002) and fewer training iterations. This strategy (New Optimization) can bring a 16.3% improvement on the CUB dataset and 1.1% on the ILSVRC2012 dataset. There are two possible reasons that the new optimization strategy leads to large improvement on the CUB dataset: 1) CUB is a fine-grained dataset whose 200 categories all belong to "bird"; 2) CUB contains only 5994 training images, which is far less than ILSVRC2012. These two factors make part domination and overfitting of the model on CUB more severe. Under this circumstance, reducing the training iterations and decrease the learning rate will greatly increase the localization performance.

On this basis, the proposed semantic mistakes tolerance module achieves a balance between enhancing the ability to distinguish foreground-background and reducing the part domination effect. Only using the semantic mistakes tolerance loss ($\mathcal{L}_{\text{SMT}}$) in training can raise the performance to 60.5% and 75.7% on the CUB and ILSVRC2012 dataset. The complete semantic mistakes tolerance module ($\mathcal{L}_{\text{SMT}}$ with top-K ACAM) and achieve 61.2% and 78.9% on these

[2] Code is available at https://github.com/gyguo/SLT-Net

Table 5. Ablation studies about network architecture of the localizer on CUB and ILSVRC2012 dataset, measured by GT-Know LOC.

Baseline		√	√	√	√	√	√
New Optimization			√	√	√	√	√
$\mathcal{L}_{\text{SMT}}$				√	√		√
top-K ACAM					√		√
$\mathcal{L}_{\text{VST}}$						√	√
GT-Know	ILSVRC	59.0	60.1	60.5	61.2	62.7	63.4
LOC (%)	CUB	58.0	74.3	75.7	78.9	81.3	85.6

two dataset. Moreover, the proposed visual stimulus tolerance ($\mathcal{L}_{\text{VST}}$) can make the optimal threshold cover more instances and achieves localization performance of 62.7% and 81.3% on ILSVRC2012 and CUB dataset, respectively. Finally, The complete method achieves 63.4% and 85.6% on the ILSVRC2012 and CUB dataset, respectively. The proposed localizer outperforms the baseline by 2.4% on ILSVRC2012 and 11.3% on CUB.

Comparison of different methods to generate pseudo bounding boxes. The learning framework of our method is similar to PSOL [42]. The main difference between POSL and our method exists in the localizer. Different PSOL that directly uses an unsupervised co-localization method DDT [39] as the localizer, we propose a learning-based approach to make the model use the classification label reasonably. In Table 6, we list the GT-Known LOC accuracy of DDT, CAM[53] and our localizer on both CUB and ILSVRC2012 dataset. We use the reported results of DDT in the original paper of PSOL, in which the resolution of the training image is 448×448. Following PSOL, we evaluate our method on various backbone networks to choose the localizer to generate pseudo bounding boxes.

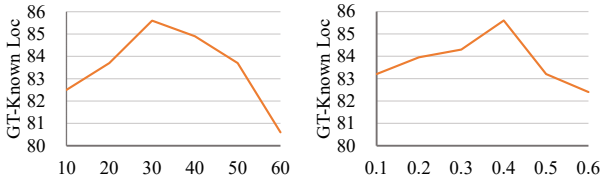
As shown in Table 6, Even though DDT can achieve a high localization performance on the CUB dataset (85.6 in terms of GT-Known LOC), but its performance on ILSVRC2012 is not good enough. Its best localization performance on ILSVRC2012 is 61.9%, which is lower than CAM (62.7%). Compared to DDT, our method brings a small performance improvement on the CUB dataset (from 84.6% to 85.6%) and lifts the localization performance of ILSVRC2012 from 61.9% to 66.7%.

Moreover, based on the performance comparison in Table 6, we choose the localizer with InceptionV3 to generate pseudo boxes for the ILSVRC2012 dataset and the localizer with VGG16 to generate pseudo boxes for the CUB dataset.

Ablation studies about hyperparameter K and α. K and α represent how the localizer tolerates the semantic mistakes. As shown in Figure 4, we chose different K and α values to study the localization performance changes with the variant semantic mistakes tolerance. For hyperparameter K, the localization accuracy first increases and reaches the largest performance at $K = 30$. This is because the model will become more tolerant to the semantic mistakes

Table 6. The GT-Known LOC accuarcy of co-localization method (DDT) and our method (Ours). The best results are highlighted in bold font and red, while the second bests are in green.

Method	ILSVRC2012	CUB
CAM-VGG16 [53]	59.0	58.0
CAM-InceptionV3 [53]	62.7	-
DDT-ResNet50 [39]	59.9	72.4
DDT-VGG16 [39]	61.4	84.6
DDT-InceptionV3 [39]	51.9	51.8
DDT-DenseNet161 [39]	61.9	78.1
Ours-DenseNet161	55.3	64.9
Ours-ResNet50	54.0	68.2
Ours-VGG16	63.4	**85.6**
Ours-InceptionV3	**65.7**	78.6



(a) K (b) α

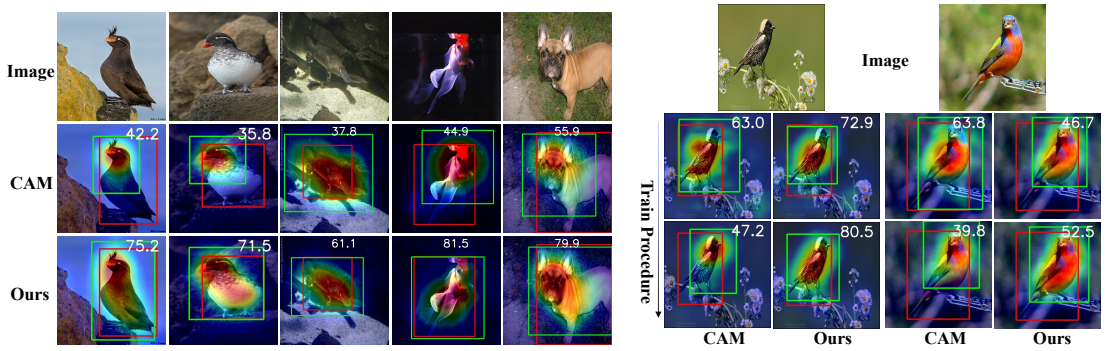
Figure 4. GT-Known LOC (%) with the changes of hyperparameters K and α on CUB dataset.

when K is larger, and the part domination problem can be alleviated. However, if the localizer is too tolerated to the semantic mistakes, the discrimination ability of the model will obviously degenerate, which will make the model can not distinguish the foreground and background very well, that is the reason why the localization performance declines when K changes from 30 to 60. Also, lager α will enhance the discrimination ability of the model, which makes the performance increase when α increases from 0.1 to 0.4. However, the performance will decline rapidly after that. This is due to when α approaches 1, the proposed semantic tolerance loss will close to the traditional classification loss, which aggravates the part domination.

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

We compare our method with state-of-the-art WSOL methods on CUB and ILSVRC2012 dataset: CAM [53], Hide-and-Seek [28], AcL [48], SPG [49], ADL [6], DANet [40], MEIL [20], PSOL [42], GC-Net [19], I2C [50].

Experiments on CUB. In Table 1 and 2, we compare our method with recent WSOL methods on both classification and localization performance. This paper changes the output size of the classifier from 1000 to 200 and initialized the remaining weights using those pre-trained from ILSVRC2012. Our method achieves classification performance of 76.6% and 76.4% in terms of Top1 CLS when the backbone is VGG16 and InceptionV3, respectively. For GT-Known LOC that reflects the localization performance, our method achieves a novel state-of-the-art performance of 87.6% based on VGG16 and achieves 86.5% when the backbone is InceptionV3. In terms of Top1-LOC, our



(a) (b)

Figure 5. Visualization comparison with the baseline CAM method. The groundtruth bounding boxes are in red, the predictions are in green, and the iou values (%) are shown in white text.

method exceeds PSOL [42] by a margin of 1.5% and 0.6% when the backbone is VGG16 and InceptionV3, respectively. Results on the CUB dataset demonstrate that the proposed method can achieve good localization performance in fine-grained images.

Experiments on ILSVRC2012. Table 3 and Table 4 report the classification and localization performance of all methods on the ILSVRC2012 dataset. For classification, we directly use the pre-trained weights provided by PyTorch [21] for the classifier. When the backbone is VGG16, we achieve 72.4% in terms of Top1 CLS, and InceptionV3 based classifier achieves 78.1% Top1 CLS.

For the localization performance, our method outperforms PSOL [42] by 3.2% and 2.4% in terms of GT-Known LOC when the backbone is VGG16 and InceptionV3, respectively. Also, our SLT-Net outperforms PSOL in terms of TOP1/Top5 LOC on both two backbones. The performance superior to PSOL [42] demonstrates the proposed learning-driven localizer performs well on large scale natural images. I2C [50] is another well-performed method that shows high performance on ILSVRC2012. When VGG16 is used as the backbone, our SLT-Net is ahead of I2C in all three metrics. When InceptionV3 is the backbone, we achieve comparable performance to I2C. Our SLT-Net outperforms I2C in terms of Top1/Top5 LOC but slightly underperforms I2C in terms of GT-Known LOC.

Results on ResNet and DenseNet. Following previous separated localization-classification method PSOL [42], we also provide the localization performance on the backbone ResNet and DenseNet in Table 7. We report the results of Top-1 LOC and GT-Known LOC.

Table 7. The localization performance on the backbone ResNet and DenseNet. We report the localization performance of PSOL and our SLT-Net.

PSOL/SLT-Net	CUB LOC		ILSVRC2012 LOC	
	Top-1	GT-Known	Top-1	GT-Known
ResNet50	70.7/72.3	-/90.7	54.0/56.2	65.4/68.5
DenseNet161	75.0/75.8	92.5/93.4	55.3/57.1	66.3/69.0

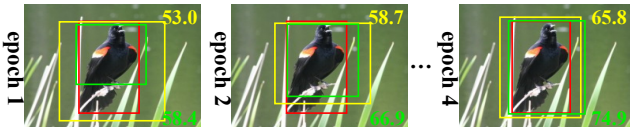


Figure 6. Visualization of groundtruth, localizer and regressor boxes in different epochs.

4.4. Visualization

In Figure 5, we give some visualization comparison between CAM with our new optimization strategy (mentioned in Table 5) and the proposed SLT-Net. Figure 5a gives the localization results on CUB and ILSVRC2012, and Figure 5b shows the change of localization results during training process, which demonstrate that our methods can significantly alleviate the part domination and over-sensitive to image diversity. Besides, in Figure 6, we show the predictions of localizer and regressor in different training epochs.

5. Conclusion

We propose a Strengthen Learning Tolerance approach (SLT-Net) to improve the localization performance of the separate localization-classification framework. We first improve the tolerance to semantic classification mistakes by reducing the loss when top-K predicted categories contain the correct label, which will alleviate the part domination problem because the localizer does not need to distinguish similar categories accurately. Moreover, to make the model less sensitive to image distribution diversity, we apply several visual transformations to the train images and match their class activation maps to that of the original image. The proposed SLT-Net can achieve 55.7% in terms of Top-1 localization accuracy, which surpasses the current state-of-the-art method.

Acknowledgement: This work was supported in part by Key R&D Program of Guangdong Province (2019B010110001), the National Science Foundation of China under Grants 61876140 and U1801265, and Research Funds for Interdisciplinary subject, NWPU.

References

[1] Wonho Bae, Junhyug Noh, and Gunhee Kim. Rethinking class activation mapping for weakly supervised object localization. In *ECCV*, 2020. 3, 5

[2] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *CVPR*, pages 2846–2854, 2016. 3

[3] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *WACV*, pages 839–847, 2018. 3

[4] Ze Chen, Zhihang Fu, Rongxin Jiang, Yaowu Chen, and Xian-Sheng Hua. Slv: Spatial likelihood voting for weakly supervised object detection. In *CVPR*, pages 12995–13004, 2020. 3

[5] Junsuk Choe, Seong Joon Oh, Seungho Lee, Sanghyuk Chun, Zeynep Akata, and Hyunjung Shim. Evaluating weakly supervised object localization methods right. In *CVPR*, pages 3133–3142, 2020. 1

[6] Junsuk Choe and Hyunjung Shim. Attention-based dropout layer for weakly supervised object localization. In *CVPR*, pages 2219–2228, 2019. 1, 3, 5, 6, 7

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 5

[8] Chuang Gan, Deng Huang, Hang Zhao, Joshua B Tenenbaum, and Antonio Torralba. Music gesture for visual sound separation. In *CVPR*, pages 10478–10487, 2020. 3

[9] Chuang Gan, Naiyan Wang, Yi Yang, Dit-Yan Yeung, and Alex G Hauptmann. Devnet: A deep event network for multimedia event detection and evidence recounting. In *CVPR*, pages 2568–2577, 2015. 1

[10] Chuang Gan, Hang Zhao, Peihao Chen, David Cox, and Antonio Torralba. Self-supervised moving vehicle tracking with stereo sound. In *ICCV*, pages 7053–7062, 2019. 3

[11] Yan Gao, Boxiao Liu, Nan Guo, Xiaochun Ye, Fang Wan, Haihang You, and Dongrui Fan. C-midn: Coupled multiple instance detection network with segmentation guidance for weakly supervised object detection. In *ICCV*, pages 9834–9843, 2019. 3

[12] Weifeng Ge, Sibei Yang, and Yizhou Yu. Multi-evidence filtering and fusion for multi-label classification, object detection and semantic segmentation based on weakly supervised learning. In *CVPR*, pages 1277–1286, 2018. 3

[13] Ross Girshick. Fast r-cnn. In *CVPR*, pages 1440–1448, 2015. 4

[14] Hao Guo, Kang Zheng, Xiaochuan Fan, Hongkai Yu, and Song Wang. Visual attention consistency under image transforms for multi-label image classification. In *CVPR*, pages 729–739, 2019. 5

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1

[16] Xiangteng He and Yuxin Peng. Weakly supervised learning of part selection model with spatial constraints for fine-grained image classification. In *AAAI*, 2017. 1

[17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, pages 4700–4708, 2017. 1

[18] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37, 2016. 1

[19] Weizeng Lu, Xi Jia, Weicheng Xie, Linlin Shen, Yicong Zhou, and Jinming Duan. Geometry constrained weakly supervised object localization. In *ECCV*, 2020. 1, 3, 5, 6, 7

[20] Jinjie Mai, Meng Yang, and Wenfeng Luo. Erasing integrated learning: A simple yet effective approach for weakly supervised object localization. In *CVPR*, pages 8766–8775, 2020. 1, 3, 5, 6, 7

[21] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS*, 2017. 6, 8

[22] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015. 1

[23] Zhongzheng Ren, Zhiding Yu, Xiaodong Yang, Ming-Yu Liu, Yong Jae Lee, Alexander G Schwing, and Jan Kautz. Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In *CVPR*, pages 10598–10607, 2020. 3

[24] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 5

[25] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626, 2017. 1, 3

[26] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR*, 2014. 3, 6

[27] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015. 1, 6

[28] Krishna Kumar Singh and Yong Jae Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *ICCV*, pages 3544–3553, 2017. 1, 3, 6, 7

[29] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *ICML*, pages 1139–1147, 2013. 6

[30] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015. 1

[31] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, pages 2818–2826, 2016. 6

[32] Peng Tang, Xinggang Wang, Song Bai, Wei Shen, Xiang Bai, Wenyu Liu, and Alan Yuille. Pcl: Proposal cluster learning for weakly supervised object detection. *TPAMI*, 42(1):176–191, 2018. 3

[33] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple instance detection network with online instance classifier refinement. In *CVPR*, pages 2843–2851, 2017. 3

[34] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 5

[35] Fang Wan, Chang Liu, Wei Ke, Xiangyang Ji, Jianbin Jiao, and Qixiang Ye. C-mil: Continuation multiple instance learning for weakly supervised object detection. In *CVPR*, pages 2199–2208, 2019. 3

[36] Fang Wan, Pengxu Wei, Zhenjun Han, Kun Fu, and Qixiang Ye. Weakly supervised object detection with correlation and part suppression. In *ICIP*, pages 3638–3642, 2016. 3

[37] Fang Wan, Pengxu Wei, Jianbin Jiao, Zhenjun Han, and Qixiang Ye. Min-entropy latent model for weakly supervised object detection. In *CVPR*, pages 1297–1306, 2018. 3

[38] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *CVPR*, pages 12275–12284, 2020. 5

[39] Xiu-Shen Wei, Chen-Lin Zhang, Jianxin Wu, Chunhua Shen, and Zhi-Hua Zhou. Unsupervised object discovery and co-localization by deep descriptor transformation. *Pattern Recognition*, 88:113–126, 2019. 3, 7

[40] Haolan Xue, Chang Liu, Fang Wan, Jianbin Jiao, Xiangyang Ji, and Qixiang Ye. Danet: Divergent activation for weakly supervised object localization. In *ICCV*, pages 6589–6598, 2019. 1, 5, 6, 7

[41] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, pages 6023–6032, 2019. 3

[42] Chen-Lin Zhang, Yun-Hao Cao, and Jianxin Wu. Rethinking the route towards weakly supervised object localization. In *CVPR*, pages 13460–13469, 2020. 1, 3, 5, 6, 7, 8

[43] Dingwen Zhang, Junwei Han, Guangyu Guo, and Long Zhao. Learning object detectors with semi-annotated weak labels. *TCSVT*, 29(12):3622–3635, 2018. 3

[44] Dingwen Zhang, Junwei Han, Le Yang, and Dong Xu. Spftn: a joint learning framework for localizing and segmenting objects in weakly labeled videos. *TPAMI*, 2018. 3

[45] Dingwen Zhang, Junwei Han, Long Zhao, and Deyu Meng. Leveraging prior-knowledge for weakly supervised object detection under a collaborative self-paced curriculum learning framework. *IJCV*, 127(4):363–380, 2019. 3

[46] Dingwen Zhang, Junwei Han, Long Zhao, and Tao Zhao. From discriminant to complete: Reinforcement searching-agent learning for weakly supervised object detection. *TNNLS*, 2020. 3

[47] Xiaopeng Zhang, Jiashi Feng, Hongkai Xiong, and Qi Tian. Zigzag learning for weakly supervised object detection. In *CVPR*, pages 4262–4270, 2018. 3

[48] Xiaolin Zhang, Yunchao Wei, Jiashi Feng, Yi Yang, and Thomas S Huang. Adversarial complementary learning for weakly supervised object localization. In *CVPR*, pages 1325–1334, 2018. 1, 3, 5, 6, 7

[49] Xiaolin Zhang, Yunchao Wei, Guoliang Kang, Yi Yang, and Thomas Huang. Self-produced guidance for weakly-supervised object localization. In *ECCV*, pages 597–613, 2018. 1, 3, 5, 6, 7

[50] Xiaolin Zhang, Yunchao Wei, and Yi Yang. Inter-image communication for weakly supervised localization. In *ECCV*, 2020. 3, 5, 6, 7, 8

[51] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *ICCV*, pages 1735–1744, 2019. 3

[52] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *ECCV*, pages 570–586, 2018. 3

[53] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016. 1, 3, 4, 5, 6, 7