Coarse2Fine: Local Consistency Aware Re-prediction for Weakly Supervised Object Localization

Yixuan Pan[1, 2], **Yao Yao**[3, 4], **Yichao Cao**[1, 2], **Chongjin Chen**[1, 2], **Xiaobo Lu**[1, 2*]

[1]School of Automation, Southeast University, Nanjing, China.
[2]Key Laboratory of Measurement and Control of Complex Systems of Engineering, Ministry of Education, Nanjing, China.
[3]Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences Shanghai, China.
[4]University of Chinese Academy of Sciences, Beijing, China.
{yixuanpan, caoyichao}@seu.edu.cn, yaoyao21@mails.ucas.ac.cn, chongjinchen@outlook.com, xblu2013@126.com

Abstract

Weakly supervised object localization aims to localize objects of interest by using only image-level labels. Existing methods generally segment the activation map by threshold to obtain mask and generate a bounding box. However, the activation map is locally inconsistent, i.e., similar neighboring pixels of the same object are not equally activated, which leads to the blurred boundary issue: the localization result is sensitive to the threshold, and the mask obtained directly from the activation map loses the fine contours of the object, making it difficult to obtain a tight bounding box. In this paper, we introduce the Local Consistency Aware Re-prediction (LCAR) framework, which aims to recover the complete fine object mask from the locally inconsistent activation map and hence obtain a tight bounding box. To this end, we propose the self-guided re-prediction module (SGRM), which employs a novel Aggregation Net with dynamic weights to replace the post-processing of threshold segmentation. To derive more reliable pseudo labels from the activation map to supervise the SGRM, we further design an affinity refinement module (ARM) that utilizes the original image feature to better align the activation map with the image contents and design a self-distillation CAM (SD-CAM) to alleviate the localizer dependence on saliency. Experiments demonstrate that our LCAR outperforms the state-of-the-art on both the CUB-200-2011 and ILSVRC datasets, achieving 95.9% and 70.7% of GT-Know localization accuracy, respectively.

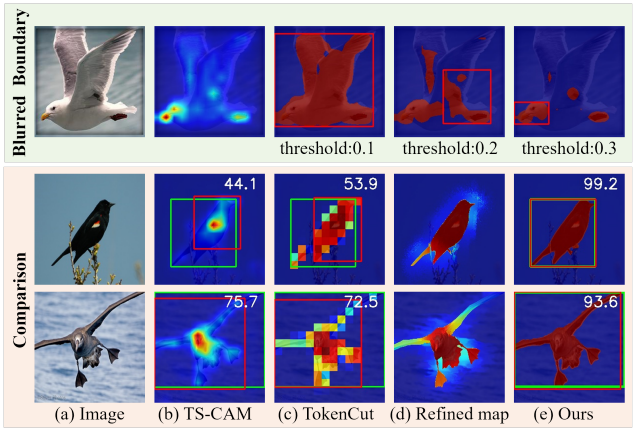

Figure 1: Blurred boundary issue and comparison of localization results: (a) Original image. (b) (c) The activation map is locally inconsistent, sensitive to threshold. (d) The output of ARM well aligns with the original appearance. (e) The output mask of LCAR covers the complete region of the object and preserves the fine contours. The IoU values are displayed in white text; the prediction and ground truth are displayed in red and green, respectively. Best viewed in color.

Introduction

Fully supervised object detection (Bochkovskiy, Wang, and Liao 2020; Carion et al. 2020) has made great advances in recent years. However, these methods rely on a large number of bounding box annotations, which are labor-intensive. Differs from fully supervised methods, weakly supervised object localization (WSOL) uses only image-level annotations to localize objects, greatly reducing the annotation cost.

Pioneering work CAM (Zhou et al. 2016) performs weighted aggregation of convolution neural network (CNN) final features to generate class activation maps for object localization, but it tends to focus on discriminative region rather than the overall region of the object. Therefore, existing methods have improved CAM from the perspectives of adversarial erasing (Zhang et al. 2018a; Choe and Shim 2019; Mai, Yang, and Luo 2020), divergent activation (Xue et al. 2019), decoupling framework (Zhang, Cao, and Wu 2020; Lu et al. 2020; Guo et al. 2021). Recent methods (Gao et al. 2021; Wang et al. 2022; Chen et al. 2022) introduce transformers to further enhance the ability to discover the complete region of the object with the global modeling capability of self-attention.

Nevertheless, the above methods focus on high-level features while ignoring the object details, which generally causes the activation maps locally inconsistent, i.e., neighboring pixels belonging to the same object with similar appearance are not activated equally, thus leading to the blurred boundary issue: the mask obtained by threshold segmentation is sensitive to the threshold and loses the fine object contours, making it difficult to get a tight bounding box. As shown in Figure 1, similar parts on the bird are not

*Corresponding author.

equally activated and the localization results are sensitive to the threshold.

Based on the above analysis, we propose the Local Consistency Aware Re-prediction (LCAR) framework for combining high-level features with the original image features to obtain fine and complete object mask. Specifically, we propose the self-guided re-prediction module (SGRM), which employs the Aggregation Net to replace the post-processing of threshold segmentation. Aggregation Net uses dynamic weights rather than shared weights for each image to find the optimal solution. We obtain pseudo labels from the activation map and design a self-supervised joint loss based on superpixels to optimize Aggregation Net. To provide more reliable pseudo labels for SGRM, we further design an affinity refinement module (ARM) that utilizes the original image feature, i.e., pixel color and position, to better align the activation map with the image contents, as shown in Figure 1(d). In addition, to alleviate the dependence on object saliency, we propose self-distillation CAM (SD-CAM) to introduce additional category information for localization. The contribution of this work can be summarized as follows.

- We propose the LCAR framework to improve localization performance by exploiting local consistency, which generates tighter bounding box.
- We introduce a general post-processing module, self-guided re-prediction module (SGRM), to replace the threshold segmentation operation. SGRM learns local consistency from superpixels and generates fine object mask from the activation map.
- We propose the affinity refinement module (ARM) to better align the attention map with image contents and introduce self-distillation CAM (SD-CAM) to provide reliable class-related localization cues.
- LCAR achieves 95.9% and 70.7% GT-Know localization accuracy on CUB-200-2011 and ILSVRC, respectively, realizing state-of-the-art performance.

Related Work

WSOL with CNN. Weakly supervised objective localization (WSOL) aims to localize objects with image-level labels solely. As a representative work of WSOL, CAM (Zhou et al. 2016) obtains a class-specific localization map by weighted aggregating the output feature of CNN. However, CAM generally only discovers the most discriminative region. Thus, some methods utilize erasing strategies to regularize the localizer, for example, HaS (Kumar Singh and Jae Lee 2017) and CutMix (Yun et al. 2019) employ random erasing, while ACoL (Zhang et al. 2018a) and ADL (Choe and Shim 2019) erase discriminative regions and guide the localizer to discover sub-discriminative regions more efficiently. PSOL (Zhang, Cao, and Wu 2020), GC-net (Lu et al. 2020), and SLT-Net (Guo et al. 2021) avoid interference between classification and localization(Choe and Shim 2019) by using a classification-localization decoupled framework. SPG (Zhang et al. 2018b) and I2C (Zhang, Wei, and Yang 2020), exploit inter-pixel correlation for discovering more complete object regions. Besides, ORNet (Xie et al. 2021) and SPOL (Wei et al. 2021) make use of shallow features in classification networks to enable localization results with clearer boundaries.

WSOL with Transformer. ViT (Dosovitskiy et al. 2020) shows that transformer can be used as an effective feature extraction model in the vision domain. Based on the ViT, DINO (Caron et al. 2021) designs self-distillation loss to encourage models to learn image feature from local to global, and its high-level feature contains explicit information that can be leveraged for semantic segmentation. TS-CAM (Gao et al. 2021) introduces vision transformer to the WSOL task for the first time, extracting long-range dependencies through transformer. TokenCut (Wang et al. 2022) proposes a unified framework for unsupervised object discovery and WSOL. Specifically, TokenCut first builds the graph with the feature extracted by DINO, then projects the graph into a low-dimensional subspace using feature decomposition, and finally segments the foreground objects by normalized graph-cut (Shi and Malik 2000). LCTR (Chen et al. 2022) employs local feature to enhance the local perception of transformer to highlight weak local response. The transformer-based methods are better at discovering global region of the object, but they ignore the rich object details and contours contained in the original features of the image, which makes the localization results sensitive to the threshold and difficult to obtain tight bounding boxes.

Local Consistency. In fields such as semantic segmentation, many methods have been proposed for mask refinement because the initial prediction results are generally coarse and locally inconsistent, i.e., neighboring pixels with similar image appearance are predicted to be different semantic objects. DenseCRF (Krähenbühl and Koltun 2011) takes full account of the contextual relationships between pixels and is often used as a post-processing for mask refinement. 1Stage (Araslanov and Roth 2020) proposes the pixel-adaptive mask refinement module based on pixel adaptive convolution (Su et al. 2019) to refine the mask leveraging local RGB information. Further, AFA (Ru et al. 2022) introduces the spatial information of pixels. Besides the pixel adaptive strategies, superpixel-based methods are often adopted to achieve local consistency. Unsupervised segmentation algorithms, including SLIC (Achanta et al. 2012) and (Felzenszwalb and Huttenlocher 2004), are commonly used to generate superpixels. Based on the over-segmentation results of SLIC, (Kanezaki 2018) obtains the image segmentation results by self-supervised merging of superpixels. SPN (Kwak, Hong, and Han 2017) proposes superpixel pooling, where superpixels are up-sampled and down-sampled as the smallest unit to preserve the object contours in the mask. (Pinheiro and Collobert 2015) introduces a superpixel smoothing prior, which simply selects the most occurring prediction in each superpixel to correct the segmentation results. For WSOL, previous methods ignore the importance of local consistency, which makes it difficult to obtain tight bounding box for each object. In this work, we make full use of the original image feature to drive the localizer learning local consistency.

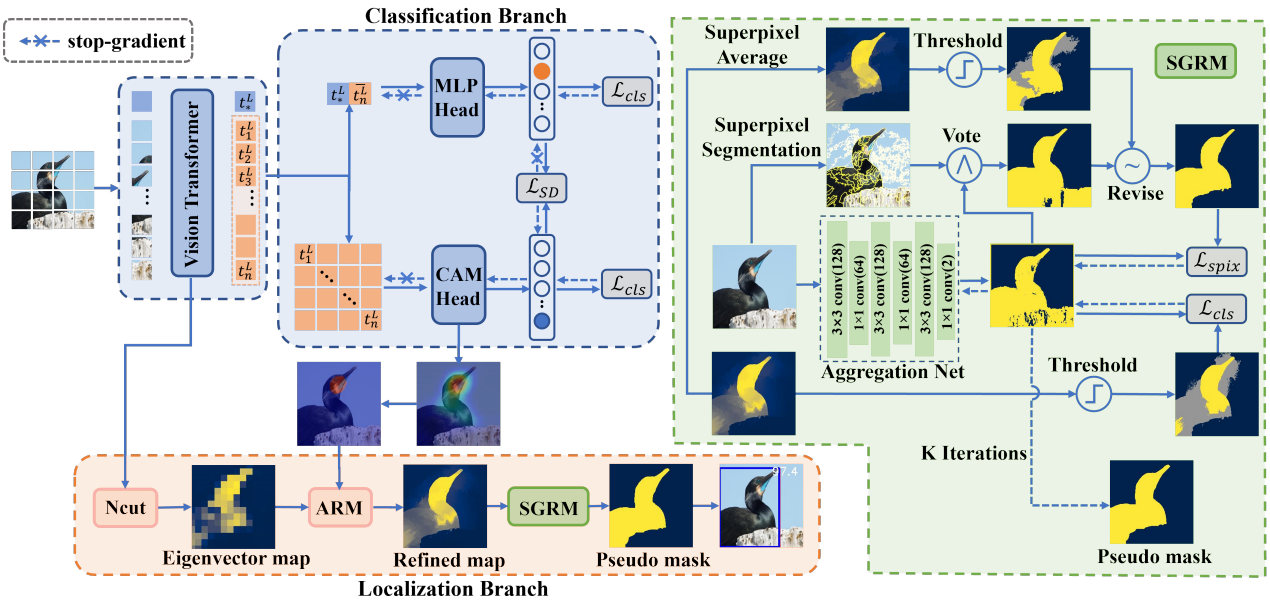

Figure 2: Framework of LCAR.

Methodology

Overview

Our proposed LCAR includes two parallel branches of classification and localization, which share the same vision transformer (Dosovitskiy et al. 2020) pretrained by DINO(Caron et al. 2021), and the weights are always frozen during the LCAR training process. The general framework of LCAR is shown in Figure 2.

Self-Distillation CAM: Obtaining Reliable Localization Seed Region

For vision transformer, the input image x with resolution $H \times W$ is initially split into $h \times w$ patches of size $k \times k$, where $h = H/k, w = W/k$. Then, $N = h \times w$ patch tokens are constructed after flattening and linear projection, which is denoted as $\{t_n^0\}_{n=1}^N$. An additional learnable class token, denoted as t_*^0, is introduced to extract the global feature. We model the sequence features consisting of class tokens and N patch tokens by the vision transformer and obtain the dependencies between tokens by exploring multi-head self-attention in each transformer block. Specifically, for the head i, the sequence of tokens will be projected by the MLP to get queries $Q_i \in \mathbb{R}^{N \times d_k}$, keys $K_i \in \mathbb{R}^{N \times d_k}$ and values $V_i \in \mathbb{R}^{N \times d_v}$, where d_k denotes the feature dimension of queries and keys, and d_v represent the feature dimension of values. The output X_i of head i is represented as:

$$X_i = \text{softmax}(\frac{Q_i K_i^\top}{\sqrt{d_k}})V_i, \tag{1}$$

where $\top$ denotes the matrix transpose operation.

The output of the transformer block X_o is derived from X_i by splicing, layer normalization and MLP projection. The output of the last transformer block X_o^L is written as $\{t_*^L, t_1^L, t_2^L, \ldots, t_N^L\}$, where t_*^L represents the encoded class token, $\{t_n^L\}_{n=1}^N$ represents the encoded patch tokens, and we treat the sequence of $N+1$ encoded tokens as the input feature of the classification branch.

In the classification branch, we initially map the 2-dimensional sequence $\{t_n^L\}_{n=1}^N$ to a 3-dimensional feature map $F \in \mathbb{R}^{h \times w \times D}$, D stands for the dimension of each token embedding. We generate CAMs according to (Zhou et al. 2016). For the feature map F and the given class c, we use the weights of the fully connected layers to weighted aggregate the feature maps in F to obtain the activation map M^c, and scale M^c to $[0, 1]$ by Min-Max normalization.Then using the background threshold β to generate the localization seed region from M^c:

$$M_{seed}^c(i) = \begin{cases} 1 & \text{if } M^c(i) > \beta \\ 0 & \text{if } M^c(i) < \beta \end{cases}. \tag{2}$$

CAM Head converts 2-dimensional sequence features into 3-dimensional feature maps as input, which only uses the information of patch tokens $\{t_n^L\}_{n=1}^N$, without effectively using the global feature class token t_*^L. We believe that this global feature is critical for finding discriminative regions. Based on the above observations, we design the self-distillation CAM (SD-CAM) to improve the reliability of localization seed region by adding a teacher model MLP Head, which implicitly exploits the global features represented by t_*^L. Specifically, denoting logits by z, the classification probability for the class i of the teacher model MLP Head output can be formulated as:

$$p_i^t = \frac{\exp(z_i^t)}{\sum_{j=1}^C \exp(z_j^t)}, \tag{3}$$

where $p^t \in \mathbb{R}^{1 \times C}$, C represents the number of classes, $z_i^t = \text{MLP}(t_*^L || \bar{t_n^L})$, $(\cdot||\cdot)$ represents the concatenation op-

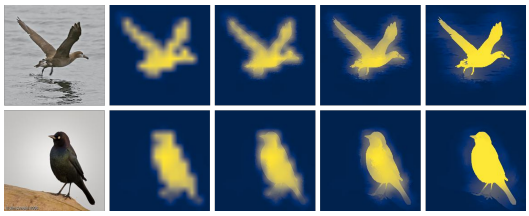

Figure 3: Visualization of the refinement process of the coarse eigenvector map with ARM.

eration, i.e., concatenating t_*^L and the mean value of patch tokens t_n^L as the input of the teacher model MLP Head. Denote the ground-truth label for image x as $y \in (1, 2, \ldots, C)$, classification loss is defined as:

$$\mathcal{L}_{cls}^{teacher} = -\log(p_y^t). \tag{4}$$

The classification probability of class i from the student model CAM Head is indicated as:

$$p_i^s = \frac{\exp\left(z_i^s\right)}{\sum_{j=1}^{C} \exp\left(z_j^s\right)}. \tag{5}$$

The CAM Head is optimized with classification loss and self-distillation loss:

$$\begin{aligned}\mathcal{L}^{student} &= \alpha \mathcal{L}_{cls}^{student} + (1-\alpha)\mathcal{L}_{SD}^{student} \\ &= -\alpha \log(p_y^s) + (1-\alpha) \sum_{i=1}^{C} S(\frac{z_i^t}{T})(\log(S(\frac{z_i^t}{T})) - \log(S(\frac{z_i^s}{T}))),\end{aligned} \tag{6}$$

where T denotes the temperature parameter to control the smoothness of the output probability, S stands for softmax operation, and α is a hyperparameter that balances the contributions of two losses. Self-distillation CAM improves the ability of the CAM Head to locate discriminative regions by implicitly utilizing global features through self-distillation learning.

Affinity Refinement Module

According to (Wang et al. 2022), we initially generate a coarse activation map from the self-attention maps of vision transformer. We build the graph with affinity matrix as edges and patch tokens $\{t_n^L\}_{n=1}^N$ as nodes, and then apply normalized graph cut (Shi and Malik 2000) to obtain the second smallest eigenvector of the generalized feature system, which is denoted as $E_l \in \mathbb{R}^{h \times w}$, i.e., the eigenvector map in Figure 2.

After obtaining the coarse eigenvector map with low resolution, we design the affinity refinement module (ARM) to obtain a fine activation map with the same resolution as the original image by utilizing the original appearance features and pixel position relationships. Concretely, inspired by (Ru et al. 2022), for the pixels at (i, j) and (k, l) positions on the input image $I \in \mathbb{R}^{h \times w \times 3}$, we define the color kernel function κ_{rgb} and the position kernel function κ_{pos} as:

$$\kappa_{rgb}^{ij,kl} = -\left(\frac{|I_{ij} - I_{kl}|}{\sigma_{rgb}^{ij}}\right)^2, \quad \kappa_{\text{pos}}^{ij,kl} = -\left(\frac{|P_{ij} - P_{kl}|}{\sigma_{pos}^{ij}}\right)^2, \tag{7}$$

where I_{xy} and P_{xy} represent the RGB information and spatial location of (x, y) pixels, σ_{rgb} and σ_{pos} denote the standard deviation of RGB and position difference, respectively. The affinity kernel function κ_{aff} is defined as a linear combination of the normalized κ_{rgb} and the normalized κ_{pos}:

$$\kappa_{aff}^{ij,kl} = \frac{\exp\left(\kappa_{rgb}^{ij,kl}\right)}{\sum\limits_{(x,y)\in\mathcal{N}} \exp\left(\kappa_{rgb}^{ij,xy}\right)} + w_p \frac{\exp\left(\kappa_{pos}^{ij,kl}\right)}{\sum\limits_{(x,y)\in\mathcal{N}} \exp\left(\kappa_{pos}^{ij,xy}\right)}, \tag{8}$$

where $\mathcal{N}$ represents the 8-way neighborhood pixels of pixel (i, j) and w_p is the position term factor. Different from (Ru et al. 2022), which only utilizes the neighbors to determine the prediction of the center pixel, we treat the result of neighborhood sampling as a correction and incremental update to the center pixel. Specifically, we get E_h with resolution $H \times W$ by bilinear interpolation from the eigenvector map E_l, and scale E_h to $[-0.5, 0.5]$ by Min-Max normalization. The result of the t-th iteration is:

$$E_{h_t}^{i,j} = \eta E_{h_{t-1}}^{i,j} + (1-\eta) \sum_{(x,y)\in\mathcal{N}} \kappa_{aff}^{ij,xy} E_{h_{t-1}}^{x,y}, \tag{9}$$

where η is used to control the update rate during iteration. After k iterations, we scale $E_{h_k} \in \mathbb{R}^{H \times W}$ to $[0, 1]$ by Min-Max normalization. However, since the eigenvector map E_l and its refinement-derived E_h do not distinguish whether the highly activated region is foreground or background, we revise E_h with the localization seed region obtained by SD-CAM. Specifically, we use the mean $\bar{E}_{h_k}$ of E_{h_k} to divide E_{h_k} into binary masks E_m, where the high activation region is assigned to 1 and the low activation region is assigned to 0. Then we separately calculate the ratio of high and low activation parts in the localization seed region M_{seed}^c to the overall high and low activation regions, denoted as R_h,R_l:

$$R_h = \frac{\sum\limits_i E_m(i) M_{seed}^C(i)}{\sum\limits_i E_m(i)}, R_l = \frac{\sum\limits_i (1 - E_m(i)) M_{seed}^C(i)}{\sum\limits_i (1 - E_m(i))}. \tag{10}$$

Further, E_{h_k} is revised so that the high activation region corresponds to the object of interest:

$$E_{h_k} = \begin{cases} E_{h_k} & \text{if} \quad R_h > R_l \\ 1 - E_{h_k} & \text{if} \quad R_h < R_l \end{cases}. \tag{11}$$

The revised E_{h_k}, i.e., the refined map in Figure 2, provides finer contours compared to the eigenvector map E_l.

Self-Guided Re-prediction Module: Re-predict Pixel-Level Mask

Although the localization results generated directly from the refined map are still threshold-sensitive, we are able to obtain reliable object regions and background regions from the refined map easily, which can be used as supervision for subsequent processing. Based on this observation, we propose the self-guided re-prediction module (SGRM), which learns semantic information from the refined map and local consistency from the superpixels to re-predict pixel-level mask.

Specifically, SGRM contains Aggregation Net, a network consisting of convolutional blocks, which is used to learn

semantic similarity and appearance similarity between pixels for dense prediction of object-background binary classification on each pixel. Based on local consistency, we generate superpixels by (Felzenszwalb and Huttenlocher 2004) for over-segmentation of the image and assume that the superpixels have consistent semantic labels inside. We average the refined map according to the superpixels so that each superpixel has the same activation value internally, denoted as E_s. Then the reliable object-background semantic label y_c is obtained from the refined map by setting the threshold, and the super-pixel semantic label y_{s^*} is obtained from E_s. The labels of the i-th pixel are:

$$y^i_{cls}=\begin{cases}0 & \text{if } E_{h_k}(i)<\gamma_{l_1}\\ 1 & \text{if } E_{h_k}(i)>\gamma_{h_1}\\ 255 & else\end{cases}, y^i_{s^*}=\begin{cases}0 & \text{if } E_s(i)<\gamma_{l_2}\\ 1 & \text{if } E_s(i)>\gamma_{h_2}\\ 255 & else\end{cases}, \tag{12}$$

where 0 denotes the pixel belongs to the background, 1 denotes the object and 255 means the pixel is not considered. We count the prediction categories of all pixels output from Aggregation Net in each superpixel region and select the majority prediction as the superpixel label in this training epoch, i.e., the vote operation in Figure 2. The voting result is marked as y_{pred}, and we revise it with y_{s^*} to get the final superpixel consistency label y_s:

$$y^i_s=\begin{cases}y^i_{s^*} & \text{if } y^i_{s^*}=1\\ 0 & \text{if } y^i_{s^*}=0\\ y^i_{pred} & else\end{cases}. \tag{13}$$

After obtaining reliable labels, we design a self-supervised joint loss to optimize Aggregation Net, which consists of classification loss and superpixel consistency loss. The superpixel consistency loss $\mathcal{L}_{spix}$ is defined as:

$$\begin{aligned}\mathcal{L}_{spix}=-\frac{1}{H\times W}\sum_{i=1}^{H\times W}[y^i_s\log O_{pred}(y^i_s)\\ +(1-y^i_s)\log(1-O_{pred}(y^i_s))]\end{aligned}, \tag{14}$$

where O_{pred} is the output of the Aggregation Net. Classification loss $\mathcal{L}_{cls}$ is a cross-entropy loss that only focuses on the reliable object-background region given by y_{cls}, including both positive and negative parts:

$$\mathcal{L}_{cls}=w_1\mathcal{L}_{cls^+}+w_2\mathcal{L}_{cls^-}, \tag{15}$$

where w_1 and w_2 balance the contributions of losses from positives and negatives. The self-supervised joint loss is defined as:

$$\mathcal{L}_{joint}=w_3\mathcal{L}_{spix}+\mathcal{L}_{cls}, \tag{16}$$

where w_3 is a hyperparameter that balances the contributions of different losses. As SGRM is trained with only one image, we can enable the simple structure of Aggregation Net to converge quickly by joint loss. Additionally, the re-prediction of an image is finished when the prediction result agrees with y_{cls} in the reliable region.

The visualization of the self-guided training process is shown in Figure 4. It can be observed that the activation value of the object region in the refined map is not consistent, after re-prediction, we obtain a mask with accurate boundaries.

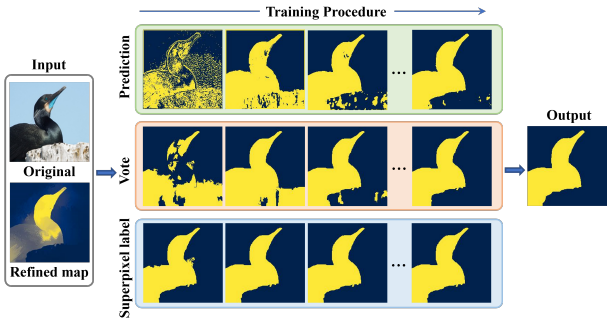

Figure 4: Self-guided training process.

Experiments

Experimental Setup

Datasets. LCAR is evaluated on two wildly used challenging WSOL benchmarks, including CUB-200-2011(Wah et al. 2011) and ILSVRC(Russakovsky et al. 2015). CUB-200-2011 is a fine-grained dataset containing 200 different bird species, consisting of 5,994 training images and 5,794 test images. For ILSVRC, we chose the subset which contains 1.2 million training images and 50,000 test images for the WSOL task. We use image-level labels for training and the bounding box annotations are solely used for evaluation.

Metrics. Following previous methods (Zhou et al. 2016; Choe and Shim 2019), we adopt the Top-1 classification accuracy (Top-1 *Cls.*), Top-1 localization accuracy (Top-1 *Loc.*) and localization accuracy with ground-truth class (GT-*k*.) as our evaluation metrics. For localization, a prediction is correct means the classification is right, while the predicted bounding box has over 50% intersection over union(IoU) with at least one of the ground-truth boxes. GT-*k*. indicates that the classification is ignored and only localization is considered. In particular, in order to evaluate the tightness of the bounding box to the object, we propose GT-*k*. with multiple IoU thresholds(GT(δ)). In contrast to GT(δ), which uses a fixed IoU threshold $\delta=0.5$, GT(δ) adopts multi-IoU $\delta=\{0.3,0.5,0.7,0.8,0.9\}$.When $\delta=\{0.3,0.5,0.7\}$, GT(δ) is equivalent to MaxBoxAccV2 (Choe et al. 2020).

Implementation Details. We adopt the ViT-S/16 model (Dosovitskiy et al. 2020) pretrained by DINO(Caron et al. 2021) as the backbone. The MLP Head in the classification branch consists of an MLP block and a linear layer. We remove the MLP block when training on ILSVRC. For the training of the classification branch, each input image is rescaled to 256×256 pixels and randomly cropped to 224×224 pixels. The input image is rescaled to 224×224 pixels directly during testing. For the self-distillation loss, we set α to 0.3 and the temperature factor T to 3. For testing, we fix the background threshold β for generating the localization seed region to 0.6; In ARM, the position term factor w_p is set to 0.01, the iterative update speed factor η is set to 0.1, and the number of iterations is fixed to 50; In SGRM, the threshold values γ_{l_1}, γ_{h_1}, γ_{l_2}, γ_{h_2} are set to 0.1, 0.6, 0.1, 0.4 respectively (γ_{h_2} is set to 0.6 in ILSVRC), the loss function weights w_1,w_2 and w_3 are fixed as 2,0.1,0.1 separately. We

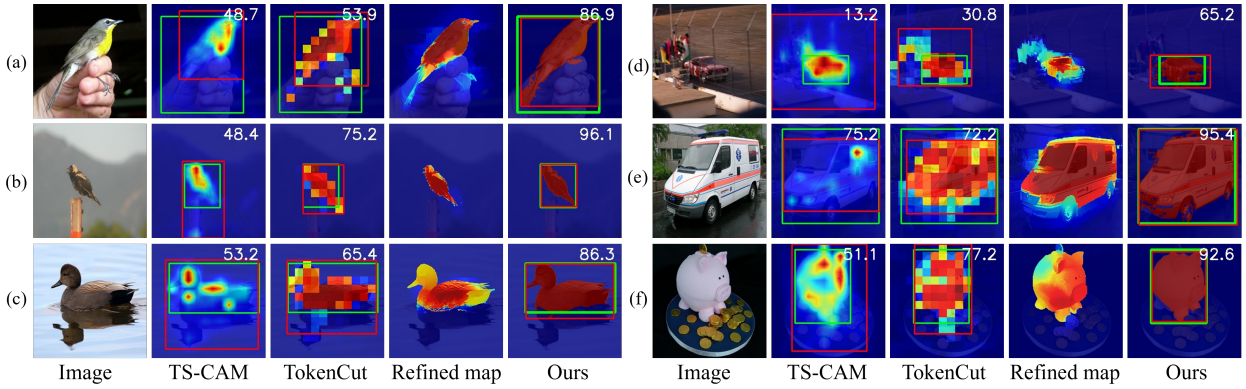

Figure 5: Comparison of different methods on CUB-200-2011 and ILSVRC. Note that the groundtruth bounding boxes are in green, the predictions are in red, and the IoU values (%) are shown in white text.

				CUB				ILSVRC			
	SD-CAM	ARM	SGRM	GT(0.5)	GT(0.7)	GT(0.8)	GT(0.9)	GT(0.5)	GT(0.7)	GT(0.8)	GT(0.9)
TokenCut*				90.7	64.2	39.5	7.9	67.7	49.6	34.7	11.2
LCAR	√			90.7	64.2	39.5	7.9	70.0	52.2	37.3	13.2
	√	√		94.4	70.8	47.6	19.1	69.3	52.8	40.3	21.6
	√		√	95.1	69.8	40.0	8.1	70.2	54.2	41.5	21.7
		√	√	95.0	79.5	61.9	27.9	68.9	53.4	42.5	24.2
	√	√	√	**95.9**	**80.6**	**62.1**	**28.9**	**70.7**	**55.5**	**44.9**	**26.7**

Table 1: Ablation Studies. Note that * denotes the re-implement method.

use SGD to optimize Aggregation Net, the momentum and learning rate are set to 0.9 and 0.05 respectively. When the percentage of prediction errors in the reliable region y_{cls} is less than 10% or reaches the maximum number of iterations (default 200), the iteration of SGRM is ended.

Ablation Studies

In this section, we conduct ablation studies to explore the effectiveness of the components in LCAR. We adopt the predictions of the MLP Head as the classification results and the ablation studies are mainly about localization. The quantitative localization results of ablation analysis are reported in Table 1. We evaluate the performance of the LCAR on both CUB-200-2011 and ILSVRC. Specifically, on the CUB200-2011 test set, LCAR increases GT(0.5) by 5.2% compared to the re-implement baseline Tokencut(Wang et al. 2022). In addition, LCAR shows a more significant improvement over the baseline when using a more strict metric. GT(δ), $\delta = \{0.7, 0.8, 0.9\}$ increase by 16.4%, 22.6% and 21.0%, respectively, which indicates that LCAR produces a much tighter localization bounding box. Similar results are provided on ILSVRC.

SD-CAM. The proposed self-distillation CAM implicitly utilizes the information of class token, which effectively improves the ability of CAM to find discriminative regions and the reliability of localization seed regions. The quantitative results in Table 1 demonstrate that the class information introduced by SD-CAM reduces the dependence of the localizer on saliency and increases GT(0.5) by 2.3% on ILSVRC. It should be noted that SD-CAM does not bring an improvement on CUB-200-2011 but improves the accuracy on ILSVRC, because there are only single instances in CUB, the foreground object can be correctly found by saliency, while multiple instances exist in ILSVRC, and relying on saliency alone will cause errors.

	GT(0.5)	GT(0.7)	GT(0.8)	GT(0.9)
TS-CAM	87.7	50.1	24.1	3.4
TS-CAM$^+$	88.8	57.0	35.3	13.8
TokenCut	91.7	67.3	43.9	9.9
TokenCut$^+$	95.0	79.5	61.9	27.9

Table 2: The effectiveness of ARM and SGRM as post-processing. Note that $^+$ denotes that ARM and SGRM are used as post-processing.

ARM. In Figure 3 we visualize the refinement process of the coarse eigenvector map with ARM. As the iterative process proceeds, the activation map has a finer object outline, and the activation level inside the object becomes higher, while the background region is effectively suppressed. As shown in Table 1, ARM significantly improves localization accuracy.

SGRM. The results in Table 1 indicate that SGRM is effective in enhancing the quality of the bounding box, with each localization metric reaching its highest when combined with ARM. To further illustrate the enhancement effect of ARM and SGRM on coarse prediction, we use both ARM and SGRM as post-processing, and the results on CUB-200-2011 are shown in Table 2, which show a significant improvement.

Methods (Yr)	Backbone	CUB				ILSVRC			
		IoU30	IoU50	IoU70	mean	IoU30	IoU50	IoU70	mean
CAM('16)	ResNet50	95.7	73.3	19.9	63.0	83.7	65.7	41.6	63.7
SPG('18)	ResNet50	92.2	68.2	20.8	60.4	83.9	65.4	40.6	63.3
ADL('19)	ResNet50	97.7	78.1	23.0	66.3	83.6	65.6	41.8	63.7
TS-CAM('21)	ViT-S/16	98.9	87.7	50.1	78.9	80.9	66.1	44.6	63.9
ViTOL('22)	ViT-B/16	96.7	80.9	39.7	72.4	**87.0**	**71.3**	49.3	69.2
LCAR(ours)	ViT-S/16	**99.3**	**95.9**	**80.6**	**91.9**	82.1	70.7	**55.5**	**69.4**

Table 3: Comparison of MaxBoxAccV2 scores on the CUB-200-2011 and ILSVRC datasets.

Methods(*Yr*)	Backbone	*Cls*	*Loc*	
		Top-1	Top-1	GT-*k*
CAM('16)	VGG16	76.6	44.2	56.0
SPG('18)	VGG16	75.5	48.9	58.9
ACoL('18)	VGG16	71.9	45.9	59.3
DANet('19)	VGG16	75.4	52.5	67.7
ADL('19)	VGG16	65.3	52.4	75.4
GC-Net('20)	VGG16	76.8	63.2	81.1
PSOL('20)	ResNet50	/	70.7	90.0
SPA('21)	VGG16	76.1	60.3	77.3
ORNET('21)	VGG16	77	67.7	86.2
SLT('21)	VGG16	76.6	67.8	87.6
FAM('21)	VGG16	77.3	69.3	89.3
CREAM('22)	InceptionV3	/	71.8	90.4
Kim *et al.*('22)	VGG16	/	70.8	93.2
TS-CAM('21)	ViT-S/16	80.3	71.3	87.7
TokenCut('22)	ViT-S/16	79.5	72.9	91.8
LCAR(ours)	ViT-S/16	**80.6**	**77.4**	**95.9**

Table 4: Comparison of localization and classification accuracy with state-of-the-art methods on CUB-200-2011.

Methods (*Yr*)	Backbone	*Cls*	*Loc*	
		Top-1	Top-1	GT-*k*
CAM('16)	InceptionV3	73.3	46.3	62.7
SPG('18)	InceptionV3	69.7	48.6	64.7
ACoL('18)	VGG16	67.5	45.8	63.0
ADL('19)	ResNet50	75.9	48.5	/
GC-Net('20)	InceptionV3	77.4	49.1	/
PSOL('20)	ResNet50	/	54.0	65.4
SPA('21)	InceptionV3	73.3	52.7	68.3
ORNet('21)	VGG16	71.6	52.1	68.3
SLT('21)	InceptionV3	**78.1**	55.7	67.6
FAM('21)	InceptionV3	77.6	55.2	68.6
CREAM('22)	ResNet50	/	55.7	69.3
Kim *et al.*('22)	ResNet50	/	53.8	69.9
TS-CAM('21)	ViT-S/16	74.3	53.4	67.6
TokenCut('22)	ViT-S/16	77.0	52.3	65.4
ViTOL('22)	ViT-S/16	71.8	53.6	70.5
LCAR(ours)	ViT-S/16	75.9	**57.1**	**70.7**

Table 5: Comparison of localization accuracy and classification accuracy with state-of-the-art methods on ILSVRC.

Comparison with State-of-the-Arts

Quantitative Comparison. To demonstrate the effectiveness of the LCAR method, we compare LCAR with the WSOL state-of-the-art methods on CUB-200-2011 and ILSVRC datasets. As shown in Table 4, LCAR achieves the highest localization and classification accuracy on CUB-200-2011, outperforming the baseline model (Wang et al. 2022) by 1.1%, 4.5%, and 4.1% on Top-1 *cls*, Top-1 *loc*, and GT-*k*., respectively. Table 5 shows the quantitative results on ILSVRC, LCAR also achieves comparable results with state-of-the-art methods on each metric. In addition, to demonstrate that the bounding boxes produced by the LCAR method are tighter to the object boundaries, we compare the MaxBoxAccV2 results of LCAR with other methods in Table 3. LCAR achieves much higher results than other methods under more strict IoU conditions. Especially, for the IoU70 metric, LCAR achieves the best results on both datasets, outperforming the best available method by 30.5% on CUB-200-2011 and by 6.2% on the more challenging ILSVRC dataset.

Visualization Comparison. Figure 5 visualizes the localization results of TS-CAM, TokenCut and our method on the CUB200-2011 and ILSVRC datasets. Compared with the other two methods, the localization results of our LCAR cover more complete object regions with tighter bounding boxes. Compared with TokenCut, LCAR can acquire object edge details more accurately under the guidance of local consistency, for example, in Figure 5(a) LCAR can better identify the details of the beak and tail of birds, in Figure 5(f) the background noise can be filtered out. Moreover, the SGRM module enables LCAR to get rid of the limitation caused by threshold, so that the bounding boxes can better fit the object tightly. For example, both TS-CAM and TokenCut in Figure 5(d) fail to locate the object because of inappropriate threshold, while LCAR can find the object accurately.

Conclusion

In this paper, we propose a novel framework LCAR for WSOL, which effectively improves localization performance by exploiting local consistency. We design a general post-processing module, self-guided re-prediction module (SGRM), to replace the threshold segmentation operation. Furthermore, we propose the affinity refinement module (ARM) which refines the activation map by utilizing the original image features and introduce self-distillation CAM (SD-CAM) to provide reliable pseudo labels for SGRM. Extensive experiments have demonstrated the effectiveness of LCAR, outperforming the localization performance of the state-of-the-art methods.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under grant 62271143. We thank the Big Data Center of Southeast University for providing facility support for the numerical calculations in this paper.

References

Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; and Süsstrunk, S. 2012. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11): 2274–2282.

Araslanov, N.; and Roth, S. 2020. Single-stage semantic segmentation from image labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4253–4262.

Bochkovskiy, A.; Wang, C.-Y.; and Liao, H.-Y. M. 2020. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *European conference on computer vision*, 213–229. Springer.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9650–9660.

Chen, Z.; Wang, C.; Wang, Y.; Jiang, G.; Shen, Y.; Tai, Y.; Wang, C.; Zhang, W.; and Cao, L. 2022. Lctr: On awakening the local continuity of transformer for weakly supervised object localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 410–418.

Choe, J.; Oh, S. J.; Lee, S.; Chun, S.; Akata, Z.; and Shim, H. 2020. Evaluating weakly supervised object localization methods right. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3133–3142.

Choe, J.; and Shim, H. 2019. Attention-based dropout layer for weakly supervised object localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2219–2228.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Felzenszwalb, P. F.; and Huttenlocher, D. P. 2004. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2): 167–181.

Gao, W.; Wan, F.; Pan, X.; Peng, Z.; Tian, Q.; Han, Z.; Zhou, B.; and Ye, Q. 2021. Ts-cam: Token semantic coupled attention map for weakly supervised object localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2886–2895.

Guo, G.; Han, J.; Wan, F.; and Zhang, D. 2021. Strengthen learning tolerance for weakly supervised object localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7403–7412.

Kanezaki, A. 2018. Unsupervised image segmentation by backpropagation. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 1543–1547. IEEE.

Krähenbühl, P.; and Koltun, V. 2011. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems*, 24.

Kumar Singh, K.; and Jae Lee, Y. 2017. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *Proceedings of the IEEE International Conference on Computer Vision*, 3524–3533.

Kwak, S.; Hong, S.; and Han, B. 2017. Weakly supervised semantic segmentation using superpixel pooling network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Lu, W.; Jia, X.; Xie, W.; Shen, L.; Zhou, Y.; and Duan, J. 2020. Geometry constrained weakly supervised object localization. In *European Conference on Computer Vision*, 481–496. Springer.

Mai, J.; Yang, M.; and Luo, W. 2020. Erasing integrated learning: A simple yet effective approach for weakly supervised object localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8766–8775.

Pinheiro, P. O.; and Collobert, R. 2015. From image-level to pixel-level labeling with convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1713–1721.

Ru, L.; Zhan, Y.; Yu, B.; and Du, B. 2022. Learning Affinity from Attention: End-to-End Weakly-Supervised Semantic Segmentation with Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16846–16855.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3): 211–252.

Shi, J.; and Malik, J. 2000. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8): 888–905.

Su, H.; Jampani, V.; Sun, D.; Gallo, O.; Learned-Miller, E.; and Kautz, J. 2019. Pixel-adaptive convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11166–11175.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset. *Computation & Neural Systems Technical Report*.

Wang, Y.; Shen, X.; Hu, S. X.; Yuan, Y.; Crowley, J. L.; and Vaufreydaz, D. 2022. Self-supervised transformers for unsupervised object discovery using normalized cut. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14543–14553.

Wei, J.; Wang, Q.; Li, Z.; Wang, S.; Zhou, S. K.; and Cui, S. 2021. Shallow feature matters for weakly supervised object localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5993–6001.

Xie, J.; Luo, C.; Zhu, X.; Jin, Z.; Lu, W.; and Shen, L. 2021. Online refinement of low-level feature based activation map for weakly supervised object localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 132–141.

Xue, H.; Liu, C.; Wan, F.; Jiao, J.; Ji, X.; and Ye, Q. 2019. Danet: Divergent activation for weakly supervised object localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6589–6598.

Yun, S.; Han, D.; Oh, S. J.; Chun, S.; Choe, J.; and Yoo, Y. 2019. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, 6023–6032.

Zhang, C.-L.; Cao, Y.-H.; and Wu, J. 2020. Rethinking the route towards weakly supervised object localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13460–13469.

Zhang, X.; Wei, Y.; Feng, J.; Yang, Y.; and Huang, T. S. 2018a. Adversarial complementary learning for weakly supervised object localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1325–1334.

Zhang, X.; Wei, Y.; Kang, G.; Yang, Y.; and Huang, T. 2018b. Self-produced guidance for weakly-supervised object localization. In *Proceedings of the European conference on computer vision (ECCV)*, 597–613.

Zhang, X.; Wei, Y.; and Yang, Y. 2020. Inter-image communication for weakly supervised localization. In *European Conference on Computer Vision*, 271–287. Springer.

Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2921–2929.