

Weakly Supervised Object Localization via Transformer with Implicit Spatial Calibration

Haotian Bai^{*[0000-0002-5693-1993]}, Ruimao Zhang^{†[0000-0001-9511-7532]}, Jiong Wang^[0000-0003-3676-2544], and Xiang Wan

Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong (Shenzhen), China

haotianwhite@outlook.com, zhangruimao@cuhk.edu.cn

Abstract. Weakly Supervised Object Localization (WSOL), which aims to localize objects by only using image-level labels, has attracted much attention because of its low annotation cost in real applications. Recent studies leverage the advantage of self-attention in visual Transformer for long-range dependency to re-active semantic regions, aiming to avoid partial activation in traditional class activation mapping (CAM). However, the long-range modeling in Transformer neglects the inherent spatial coherence of the object, and it usually diffuses the semantic-aware regions far from the object boundary, making localization results significantly larger or far smaller. To address such an issue, we introduce a simple yet effective Spatial Calibration Module (SCM) for accurate WSOL, incorporating semantic similarities of patch tokens and their spatial relationships into a unified diffusion model. Specifically, we introduce a learnable parameter to dynamically adjust the semantic correlations and spatial context intensities for effective information propagation. In practice, SCM is designed as an external module of Transformer, and can be removed during inference to reduce the computation cost. The object-sensitive localization ability is implicitly embedded into the Transformer encoder through optimization in the training phase. It enables the generated attention maps to capture the sharper object boundaries and filter the object-irrelevant background area. Extensive experimental results demonstrate the effectiveness of the proposed method, which significantly outperforms its counterpart TS-CAM on both CUB-200 and ImageNet-1K benchmarks. The code is available at <https://github.com/164140757/SCM>.

Keywords: Weakly Supervised Object Localization, Image Context Modeling, Class Activation Mapping, Transformer, Semantic Propagation

1 Introduction

Weakly supervised object localization (WSOL), which learns to localize objects by only using image-level labels, has attracted much attention recently for its

^{*} Research done when Haotian Bai was a Research Assistant at Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong (Shenzhen).

[†] Corresponding Author

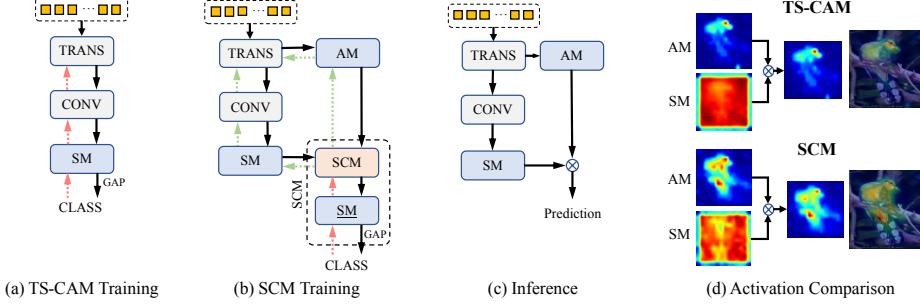


Fig. 1: Transformer-based localization pipelines in WSOL. The dashed arrows indicate the module parameters update during backpropagation. (a) TS-CAM [9]: the training pipeline encodes the feature maps into semantic maps (SM) through a convolution head, then applies a GAP to receive gradients from the image-label supervision. (b) SCM(Ours): our training pipeline incorporates external SCM to produce new semantic maps SM refined with the learned spatial and semantic correlation. Then it updates the Transformer backbone through backpropagation to obtain better attention maps and semantic representations for WOLS. (c) Inference: SCM is dropped out, and we couple attention maps (AM) and SM just like TS-CAM for final localization prediction. (d) Comparison of AM, SM, and final activation maps of TS-CAM and proposed SCM.

low annotation cost. The representative study of WSOL, Class Activation Map (CAM) [39] generates localization results using features from the last convolutional layer. However, the model trained for classification usually focuses on the discriminative regions, resulting insufficient activation for object localization. To solve such an issue, there are many CNN-based methods have been proposed in the literature, including regularization [36,33,20,31], adversarial training [5,20,36], and divergent activation [28,33,34], but the CNN’s inherent limitation of local activation dampens their performance. Although discriminative activation is optimal for minimizing image classification loss, it suffers from the inability to capture object boundaries precisely.

Recently, visual Transformer has succeeded in computer vision due to its superior ability to capture long-range feature dependency. Vision Transformer [27] splits an input image into patches with the positional embedding, then constructs a sequence of tokens as its visual representation. The self-attention mechanism enables Transformer to learn long-range semantic correlations, which is pivotal for object localization. A representative study is Token Semantic Coupled Attention Map (TS-CAM) [9] which replaces traditional CNN with Transformer and takes full advantage of long-range dependencies to solve the partial activation problem. It localizes objects by semantic-awarded attention maps from patch tokens. However, we argue that only using a Transformer is not an optimal choice in practice. Firstly, Transformer attends to long-range global dependency while inevitably it cannot capture local structure well, which is critical in describing the boundaries of objects. In addition, Transformer splits images into discrete patches. Thus it may not attend to the inherent spatial coherence of objects,

which makes it unable to predict the complete activation. As shown in Fig.1(d), the activation map obtained from TS-CAM captures the global structure. Still, it concentrates in a small semantic-rich region like the bird’s upper body, failing to solve partial activation completely. Furthermore, we observe that the fur has no abrupt change in neighboring space, and its semantic context may favor propagating the activated regions to provide a more accurate result covering the whole body.

Inspired by this potential continuity, we propose a novel external module named Spatial Calibration Module (SCM), tailored for Transformers to produce activation maps with sharper boundaries. As shown in Fig.1(a)-(b), instead of directly applying Global Average Pooling (GAP) on semantic maps to calculate loss as TS-CAM [9], we insert an external SCM to refine both semantic and attention maps and then use the calibrated features to calculate the semantic loss. Precisely, it implicitly calibrates attention representation of Transformer and produces more meaningful activation maps to cover functional areas based on spatial and contextual coherence. Our core design, a unified diffusion model, is introduced to incorporate semantic similarities of patch tokens and their local spatial relations during training. While in the inference phase, SCM can be dropped out to maintain the model’s simplicity, as shown in Fig.1(c). Then, we use the calibrated Transformer backbone to predict the localization results by coupling SM and AM. The main contributions of this paper are as follows:

1. We propose a novel spatial calibration module (SCM) as an external Transformer module to solve the partial activation problem in WSOL by leveraging the spatial correlation. Specifically, SCM is designed to optimize Transformers implicitly and will be dropped out during inference.
2. We propose a novel information propagation methodology that provides a flexible way to integrate spatial and semantic relationships to enlarge the semantic-rich regions and cover objects completely. In practice, we introduce learnable parameters to adjust the diffusion range and filter the noise dynamically for flexible control and better adaptability.
3. Extensive experiments demonstrate that the proposed framework outperforms its counterparts in the two challenging WSOL benchmarks.

2 Related Work

2.1 Weakly Supervised Object Localization.

The weakly supervised object localization aims to localize objects by solely image-level labels. The seminar work CAM [39] demonstrates the effectiveness of localizing objects using feature maps from CNNs trained initially for classification. Despite its simplicity, CAM-based methods suffer from limited discriminative regions, which cannot cover objects completely. The field has focused on how to expand the activation with various attempts. Firstly, the dropout strategy is proposed to guide the model to attend to more significant regions. For instance, HaS [28] hides patches in training images randomly to force the

network to seek other relevant parts; CutMix [34] adopts the same way to drop out patches but further augment the area of the patches with ground-truth labels to reduce information loss. Similarly, ADL [6] adopts an importance map to maintain the informative regions' classification power. Instead of dropping out patches, people leverage the pixels correlations to fulfill objects as they often share similar patterns. SPG [37] learns to sense more areas with similar distribution and expand the attention scope. I²C [38] exploits inter-and-cross images' pixel-level consistency to improve the quality of localization maps. Furthermore, the predicted masks can be enhanced to become complete. GC-Net [18] highlights tight geometric shapes to fit the masks. SPOL [30] fuses shallow features and deep features from CNN that filter the background noise and generates sharp boundaries.

Instead of applying only CNN as the backbone for WSOL, Transformer can be another candidate to alleviate the problem of partial activation as it captures long-range feature dependency. A recent study TS-CAM [9] utilizes attention maps from patches coupled with reallocated semantics to predict localization maps, surpassing most of its CNN counterparts in WSOL. Recent work LCTR [2] adopted a similar framework with Transformer while inserting their tailored module in each Transformer block to strengthen the global features. However, we observe that using Transformer alone cannot solve the partial activation completely as it fails to capture the local structure and ignores spatial coherence. What is more, it is cumbersome to insert a module for each Transformer block like LCTR [2]. To address the issue, we propose a simple external module termed spatial calibration module (SCM) that calibrates Transformer by incorporating spatial and semantic relations to provide more complete feature maps and erase background noise.

2.2 Graph Diffusion.

Pixels in natural images generally exhibit strong correlation, and constructing graph structure to capture such relationships has attracted much attention. In semantic segmentation, studies like [16,15] build graphs on images to obtain contextual information and long-term dependencies to model label distribution jointly. In image prepossessing, Gene *et.al* [3] analyses graphs constructed from 2D images in spectral and succeeds in many traditional processing areas, including image compression, restoration filtering, and segmentation. The graph structure enables many classic graph algorithms and leads to new insights and understanding of image properties.

Similarly, in WSOL, the limited activation regions share semantic coherence with neighboring locations, making it possible to expand the area by information flow to cover objects precisely. In our study, we revise the classic Graph Diffusion Kernel (GDK) algorithm [13] to infer complete pseudo masks based on partial activation results. GDK is initially adopted in graph analysis like social networks [1], search engines [19], and biology [25] to inference pathway membership in genetic interaction networks. GDK's strategy to explore graphs via random walk

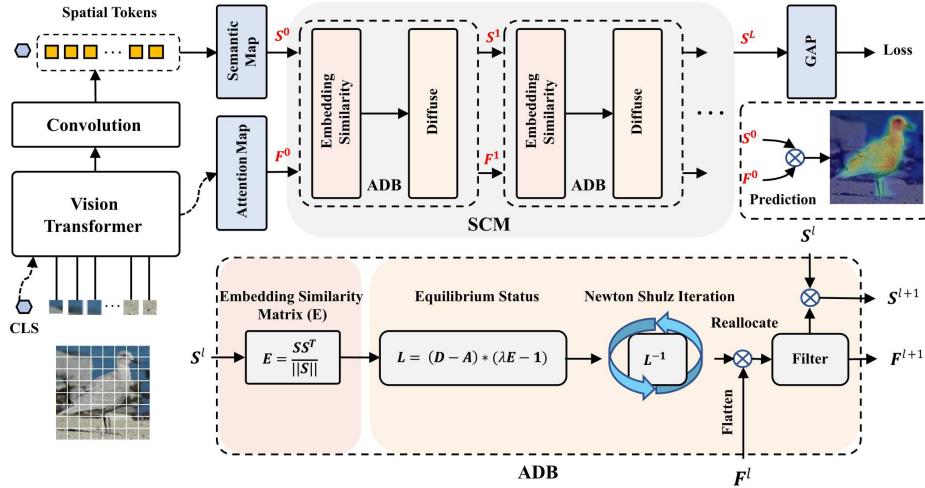


Fig. 2: The overall framework consists of two parts. (Left) Vision Transformer provides the original attention map F_0 and semantic map S_0 , (Right) They are dynamically adjusted by stacked activation diffusion blocks (ADBs). The detail of the layer design is shown at the bottom-right corner (the residual connections for F_l and S_l are omitted for simplicity). Once model optimized, F_0 and S_0 are directly element-wise multiplied for final prediction.

inspires us to modify it to incorporate information from the image context, enabling dynamical adjustment by semantic similarity.

3 Methodology

This section describes the Spatial Calibration Module (SCM), which is built by stacking multiple activation diffusion blocks (ADB). ADB consists of several submodules, including semantic similarity estimation, activation diffusion, diffuse matrix approximation, and dynamic filtering. At the end of the section, we show how to predict the final localization results by using the proposed framework during the inference.

3.1 Overall Architecture

In WSOL, the attention maps from models trained on image-level labels mainly concentrate on discriminative parts, which fail to cover the whole objects. Our proposed SCM aims to diffuse activation at small areas outwards to alleviate the partial activation problem in WSOL. In a broad view, the whole framework is supervised by image-level labels during training. As shown in Fig.1(b), Transformer learns to calibrate both attention maps and semantic maps through the semantic loss from SCM implicitly. To infer the prediction, as described in Fig.1(c), we drop SCM and use the element-wise product of revised maps to localize objects.

As shown in Fig.2, an input image is split into $N = H \times W$ patches with each represented as a token, where (H, W) is the patch resolution. After grouping these patch tokens and CLS token into a sequence, we send it into I cascaded Transformer blocks for further representation learning. Similar as TS-CAM [9], to build the initial attention map $\mathbf{F}^0 \in \mathbb{R}^{H \times W}$, the self-attention matrix $\mathbf{W}_i \in \mathbb{R}^{(N+1) \times (N+1)}$ at i^{th} layer is averaged over the multiple self-attention heads. Denote $\mathbf{M}_i \in \mathbb{R}^{H \times W}$ as attention weights that corresponds to the class token in \mathbf{W}_i , we average $\{\mathbf{M}_i\}_{i=1}^I$ across all intermediate layers to get the attention map \mathbf{F}^0 of Transformer.

$$\mathbf{F}^0 = \frac{1}{I} \sum_{i=1}^I \mathbf{M}_i \quad (1)$$

To obtain the semantic map $\mathbf{S}^0 \in \mathbb{R}^{H \times W \times C}$, where C denotes the number of categories, we extract all spatial tokens $\{\mathbf{t}_n\}_{n=1}^N$ from the last Transformer layer and then encode them by a convolution head,

$$\mathbf{S}^0 = \text{reshape}(\mathbf{t}_1 \dots \mathbf{t}_N) * \mathbf{k} \quad (2)$$

where $*$ is the convolution operation, \mathbf{k} is a 3×3 convolution kernel, and $\text{reshape}(\cdot)$ is an operation that converts a sequence of tokens into 2D feature maps. Then we send both \mathbf{F}^0 and \mathbf{S}^0 into SCM to refine them.

As illustrated in Fig.2, for the l^{th} ADB, denote \mathbf{S}^l and \mathbf{F}^l as the inputs, and \mathbf{S}^{l+1} and \mathbf{F}^{l+1} as the outputs. Firstly, to guide the propagation, we estimate embedding similarity \mathbf{E} between pairs of patches in \mathbf{S}^l . To enlarge activation \mathbf{F}^l , we apply \mathbf{E} to diffuse \mathbf{F}^l towards the equilibrium status indicated by the inverse of Laplacian matrix \mathbf{L}^l . In practice, we re-activate \mathbf{F}^l by approximating $(\mathbf{L}^l)^{-1}$ with Newton Shulz Iteration. Afterward, a dynamic filtering module is applied to remove over-diffused parts. Finally, the refined \mathbf{F}^l updates \mathbf{S}^l via an element-wise multiplication.

In general, by stacking multiple ADBs, the intensity of both maps is dynamically adjusted to balance semantic and spatial features. In the training phase, we apply GAP to \mathbf{S}^L to get classification logits and calculate semantic loss with the ground truth. During inference, SCM will be dropped out, and the element-wise product of newly extracted \mathbf{F}^0 and \mathbf{S}^0 is used to obtain the localization result.

3.2 Activation Diffusion Block

In this subsection, we dive into Activation Diffusion Block (ADB). Under the assumption of continuity of visual content, we calculate the semantic and spatial relationships of patches in \mathbf{S}^L , then diffuse it outwards dynamically to alleviate the partial activation problem in WSOL.

Semantic Similarity Estimation. Within the l^{th} activation diffusion block, $l \in \{1, 2, \dots, L\}$, we need semantic and spatial relationships between any pair of

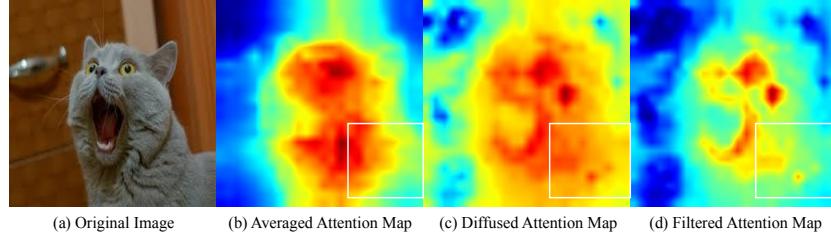


Fig. 3: Illustration of activation diffusion pipeline with a hand-crafted example. (a) Input image. (b) Original Transformer’s attention map. (c) Diffused attention map. (d) Filtered attention map. As the spatial coherence is embedded into the attention map via our SCM, the obtained attention map by using proposed method captures a complete object boundary with less noise.

patches for propagation. To achieve it, we construct an undirected graph with each \mathbf{v}_i^l connected with its first-order neighbors. Please refer to Fig.5 at the Appendix for details. Given token representation of S^l , we build an N -node graph G^l . Denote the i^{th} node as $\mathbf{v}_i^l \in \mathbb{R}^Q$. Then, we can infer the semantic similarity \mathbf{E}^l , where the specific element $\mathbf{E}_{i,j}^l$ is defined as the cosine distance between \mathbf{v}_i^l and \mathbf{v}_j^l :

$$\mathbf{E}_{i,j}^l = \frac{\mathbf{v}_i^l(\mathbf{v}_j^l)^\top}{\|\mathbf{v}_i^l\| \|\mathbf{v}_j^l\|} \quad (3)$$

where \mathbf{v}_i^l and \mathbf{v}_j^l are flattened vectors, and the larger value $\mathbf{E}_{i,j}^l$ denotes the higher similarity shared by \mathbf{v}_i^l and \mathbf{v}_j^l .

Activation Diffusion. To present spatial relationship, we define a binary adjacency matrix $\mathbf{A}^l \in \mathbb{R}^{N \times N}$, whose element $\mathbf{A}_{i,j}^l$ indicates whether \mathbf{v}_i^l and \mathbf{v}_j^l are connected. We further introduce a diagonal degree matrix $\mathbf{D}^l \in \mathbb{R}^{N \times N}$, where $D_{i,i}^l$ corresponds to the summation of all the degrees related to \mathbf{v}_i^l . Then, we obtain Laplacian matrix $\hat{\mathbf{L}}^l = \mathbf{D}^l - \mathbf{A}^l$, with each element $(\mathbf{L}^l)_{i,j}^{-1}$ describes the correlation of \mathbf{v}_i^l and \mathbf{v}_j^l at the equilibrium status.

Recent studies [16,15,8] on graph representation inspire us that the inverse of the Laplacian matrix leads to the global diffusion, which allows each unit to communicate with the rest. To enhance the diffusion with semantic relationships, we incorporate $\hat{\mathbf{L}}^l$ with node contextual information \mathbf{E}^l . Intuitively, we take advantage of the spatial connectivity and semantic coherence to split the tokens into the semantic-awarded foreground objects and the background environment. In practice, we use a learnable parameter λ to dynamically adjust the semantic intensity, which makes the diffusion process more flexible and easier to fit various situations. The Laplacian matrix \mathbf{L}^l with semantics is defined as,

$$\mathbf{L}^l = (\mathbf{D}^l - \mathbf{A}^l) \odot (\lambda \mathbf{E}^l - \mathbf{1}) \quad (4)$$

where \odot represents element-wise multiplication, and $\mathbf{1}$ denotes the information flow exchange with neighboring vertexes. $(\mathbf{D}^l - \mathbf{A}^l)$ denotes the spatial connectivity, $(\lambda \mathbf{E}^l - \mathbf{1})$ represents the semantic coherence, and \odot incorporates them for diffusion. Please refer to Appendix for full details of Eqn.(4). After the global propagation, the reallocated activation score map can be calculated as follows,

$$\mathbf{F}^{l+1} = (\mathbf{L}^l)^{-1} \Gamma(\mathbf{F}^l) \quad (5)$$

where \mathbf{F}^{l+1} is the output re-allocated attention map and Γ is a flattening operation that reshapes \mathbf{F}^l into a patch sequence.

Diffuse Matrix Approximation. In practice, directly using $(\mathbf{L}^l)^{-1}$ may be impractical since \mathbf{L}^l is not guaranteed to be positive-definite and its inverse may not exist. Meanwhile, as observed in our initial experiments, directly applying the inverse produced unwanted artifacts. To deal with the problems, we exploit Newton Schulz Iteration [23,22] to solve $(\mathbf{L}^l)^{-1}$ to approximate the global diffusion result,

$$\begin{aligned} X_0 &= \alpha(\mathbf{L}^l)^\top \\ X_{p+1} &= X_p(2\mathbf{I} - \mathbf{L}^l X_p), \end{aligned} \quad (6)$$

where X_0 is initialized as $(\mathbf{L}^l)^\top$ multiplied by a small constant value α . The subscript p denotes the number of iterations, and \mathbf{I} is the identity matrix. As discussed above, we only need $(\mathbf{L}^l)^{-1}$ to thrust propagation instead of obtaining the equilibrium result, so we just iterate the Eqn.(6) for p times then take the approximated $(\mathbf{L}^l)^{-1}$ back to Eqn.(5). Then we obtain the diffused activation of \mathbf{F}^l , which is visualized in Fig.3(c). We can see that diffusion has redistributed the averaged attention map with more boundary details, such as the *ear* and the *mouth*, which are beneficial for final object localization.

Dynamic Filtering. As depicted in Fig.3(c), we found that the reallocated score map \mathbf{F}^{l+1} provides a sharper boundary, but there is a side-effect that it diffuses the activation out of object boundaries, which may make the unnecessary background context back into \mathbf{S}^{l+1} or result in over-estimation of bounding box. Therefore, we propose a soft-threshold filter, depicted as Eqn.(7), to increase density contrast between the objects and the surrounding background to depress the outside noise.

$$\mathcal{T}(\mathbf{F}^l, \beta) = \beta \cdot \text{tanhShrink}\left(\frac{\mathbf{F}^l}{\beta}\right) \quad (7)$$

where $\beta \in (0, 1)$ is a threshold parameter for more flexible control. \mathcal{T} denotes a soft-threshold function, and $\text{tanhShrink}(x) = x - \tanh(x)$ is used to depress activation under β . Then $\mathbf{S}^{l+1} = \mathbf{S}^l \odot \mathcal{T}(\mathbf{F}^l, \beta)$. As shown in Fig.3(d), the filter operation removes noise and provides sharper contrast.

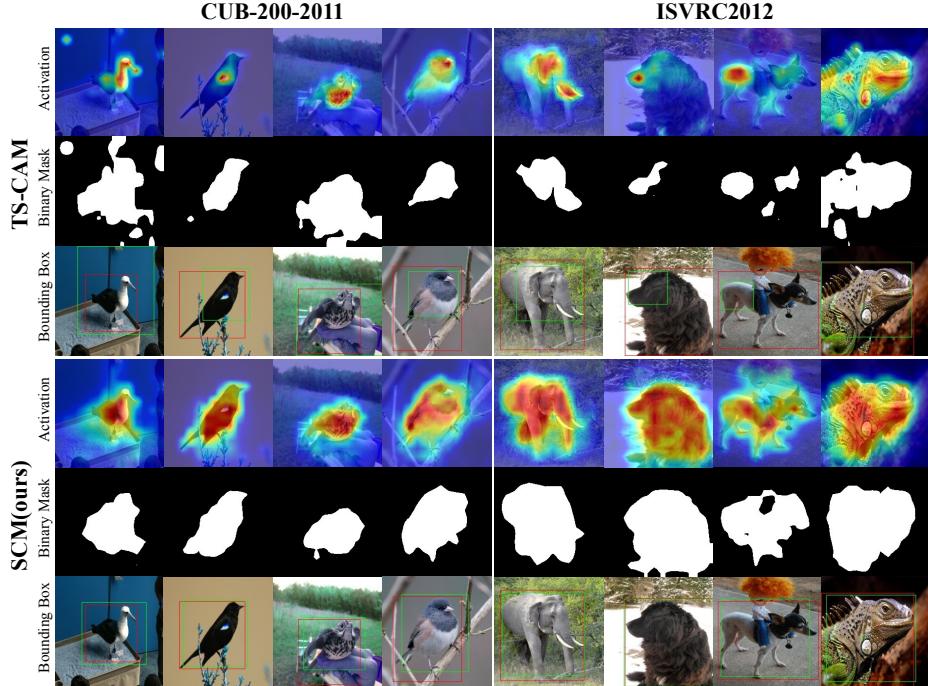


Fig. 4: Visual comparison of TS-CAM and SCM on 4 samples from CUB-200-2011 and ISVRC2012. Here we use three rows for each method to show activation maps, binary map predictions, and bounding box predictions, respectively. The threshold value γ is set to be the optimal values proposed in TS-CAM and SCM.

3.3 Prediction

After optimizing the model through backpropagation, the calibrated Transformer can generate the object-boundary-aware activation maps. Thus, we drop SCM during inference to obtain the final bounding box. Specifically, the bounding box prediction is generated by coupling \mathbf{S}^0 and \mathbf{F}^0 as depicted in Fig.2. As $\mathbf{S}^0 \in \mathbb{R}^{H \times W \times C}$ is a C -channel 2D semantic map, each channel represents an activation map for a specific class c . To obtain the prediction from score maps, we carry out the following procedures: (1) Pass \mathbf{S}^0 through a GAP to calculate classification scores. (2) Select i^{th} map $\mathbf{S}_i^0 \in \mathbb{R}^{H \times W}$ corresponding to the highest classification score from \mathbf{S}^0 . (3) Calculate the element-wise product $\mathbf{F}^0 \odot \mathbf{S}_i^0$. The coupled result is then up-sampled to the same size as the input for bounding box prediction.

4 Experiments

4.1 Experiment Settings

Datasets. We evaluate SCM on two commonly used benchmarks, CUB-200-2011 [32] and ILSVRC2012 [26]. CUB-200-2011 is an image dataset with photos

of 200 bird species, containing a training set of 5,994 images and a test set of 5,794 images. ILSVRC contains about 1.2 million images with 1,000 categories for training and 50,000 images for validation. Our SCM is trained on the training set and evaluated on the validation set from which we only use the bounding box annotations for evaluation.

Evaluation Metrics. We evaluate the performance by the commonly used metric GT-Known and save models with the best performance. For GT-Known, a bounding box prediction is positive if its Intersection-over-Union (IoU) δ with at least one of the ground truth boxes is over 50% . Furthermore, for a fair comparison with previous works, we apply the commonly reported Top1/5 Localization Accuracy(Loc Acc) and Classification Accuracy(Cls Acc). Compared with GT-Known, Loc Acc requires the correct classification result besides the condition of GT-Known. Please refer to the appendix for more strict measures like MaxboxAccV1 and MaxboxAccV2 as recommended by [4] to evaluate localization performance only.

Implementation details. The Transformer module is built upon the Deit [29] pretrained on ILSVRC. In detail, we initialize λ , β in ABDs to constant values (1 and 0.5 respectively), and choose $p = 4$ and $\alpha = 0.002$ in Eqn.(6). For input images, each sample is re-scaled to a size of 256×256 , then randomly cropped to 224×224 . The MLP head in the pretrained Transformer is replaced by a 2D convolution head with kernel size of 3, stride of 1, and padding of 1 to encode feature maps into semantic maps S^0 (200 output units for CUB-200-2011, and 1000 for ILSVRC). The new head is initialized with He’s approach [11]. During training, we use AdamW [17] with $\epsilon = 1e^{-8}$, $\beta_1 = 0.9$, $\beta_2 = 0.99$ and weight decay of 5e-4. On CUB-200-2011, the training lasts 30 epochs with an initial learning rate of 5e-5 and batch size of 256. On ILSVRC, the training procedure carries out 20 epochs with a learning rate of 1e-6 and batch size of 512. We measure model performance on the validation set after every epoch. At last, we save the parameters with the best GT-Known performance on the validation set.

4.2 Performance

To demonstrate the effectiveness of the proposed SCM, we compare it against previous methods on CUB-200-2011 and ILSVRC2012 in Table.1. From GT-Known in CUB, SCM outperforms baseline method TS-CAM [9] with a large margin, yielding GT-known 96.6% with a performance gain of 8.9%. Compared with other CNN counterparts, SCM is competitive and outperforms the state-of-the-art SPOL [30] using only about 24% parameters. As for ILSVRC, SCM surpasses TS-CAM by 1.2% on GT-Known and 5.1% on Top-1 Loc Acc and is competitive against SPOL built on the multi-stage CNN models. Compared with SPOL, SCM has the following advantages, (1)**Simple**: SPOL produces semantic maps and attention maps on two different modules separately, while SCM is only

Table 1: Comparison of SCM with state-of-the-art methods in both classification and localization on CUB [32] and ILSVRC [26] test set. The column Params indicates the number of parameters in backbone on which models are built. Values in bracket show improvement of our method compared with TS-CAM [9]. GT-K. stands for ground truth known.

Model	Backbone	Params(M)	CUB						ILSVRC					
			Cls Acc		Loc Acc			Cls Acc		Loc Acc				
			Top-1	Top-5	Top-1	Top-5	GT-K.	Top-1	Top-5	Top-1	Top-5	GT-K.		
CAM[39]	VGG16	138	-	-	34.4	-	-	68.8	88.6	42.8	54.9	-		
ACoL[36]	VGG16	138	71.9	-	45.9	61.0	-	67.5	88.0	45.8	63.3	-		
MEIL[20]	VGG16	138	74.8	-	57.5	-	-	73.3	-	49.5	-	-		
SPG[37]	InceptionV3	24	-	-	46.6	59.4	-	84.5	97.3	56.1	70.6	64.7		
I ² C[38]	InceptionV3	24	-	-	55.9	68.3	72.6	73.3	91.6	53.1	64.1	68.5		
GC-Net[18]	GoogleNet	6.8	76.8	92.3	63.2	75.5	-	77.4	93.6	49.1	58.1	-		
ADL[6]	ResNet50-SE	28	80.3	-	62.3	-	-	75.9	-	48.5	-	-		
BGC[12]	ResNet50	25.6	-	-	53.8	65.8	69.9	-	-	53.8	65.8	69.9		
PDM[21]	ResNet50	25.6	81.3	-	54.4	65.5	69.6	75.6	91.6	54.4	65.5	69.6		
LCTR[2]	Deit-S	22	85.0	97.1	79.2	89.9	92.4	77.1	93.4	56.1	65.8	68.7		
TS-CAM[9]	Deit-S	22	80.3	94.8	71.3	83.8	87.7	74.3	82.1	53.4	64.3	67.6		
SCM(ours)	Deit-S	22	78.5	94.5	76.4(5.1↑)	91.6(7.8↑)	96.6(8.9↑)	76.7(2.4↑)	93.0(10.9↑)	56.1(2.7↑)	66.4(2.1↑)	68.8(1.2↑)		
PSOL[35]	DenseNet161 + EfficientNet-B7	95.0	-	-	77.4	89.5	93.0	-	-	56.4	66.5	69.0		
SPOL[35]	ResNet50 + EfficientNet-B7	91.6	-	-	80.1	93.4	96.5	-	-	59.1	67.2	69.0		

* CNN-based models are listed above. Transformer-based models are given at the center. Both PSOL [35] and SPOL [30] are composed of multiple-stage models are listed below. The best performance is shown as **bold** for CNN-based, Transformer, and multi-stage models, respectively.

finetuned on a single backbone. (2) **Light-weighted**: SPOL is built on a multi-stage model with huge parameters, while SCM is built on a small Transformer with only about 24% parameters of the former. (3) **Convenient**: SPOL has to infer the prediction with the complex network design, but SCM is dropped out during the inference stage. Furthermore, compared with the recent Transformer-based works like LCTR [2], with the same backbone Deit-S, we surpass it by a large margin 4.2% in terms of GT-Known in CUB and obtain comparable performance on Loc Acc for both CUB and ISVRC. We achieve this without additional parameters during inference, while other recent proposed methods add carefully designed modules or processes to improve the performance. The models are saved with the best GT-Known performance and achieve satisfactory Loc Acc and Cls Acc. Please refer to Sec.4.3 for more details.

The visual comparison of SCM and TS-CAM is shown in Fig.4. We observe that TS-CAM preserves the global structure but still suffers from the partial activation problem that degrades its localization ability. Specifically, it cannot predict a complete component from the activation map. We notice that minor and sporadic artifacts appear on the binary threshold maps, and most of them include half parts of the objects. After adding SCM as a simple external adaptor, the masks become integral and accurate, so we believe that SCM is necessary for Transformers to find their niche in WSOL.

4.3 Ablation Study

In this section, we first illustrate the trade-off between localization and classification given the pre-determined backbone. Then we explore why SCM can reallocate and enlarge activation from two perspectives. Specifically, we show the visual results of both semantic maps S^l and attention maps F^l across all layers, and analyze them with the learnable parameters' trend during training. Next, we illustrate the influence of module scale by stacking a different number of ADBs. At last, we apply SCM to other Transformers like ViT [27], and Conformer [10] to prove SCM's adaptability. If not mentioned specifically, We carry

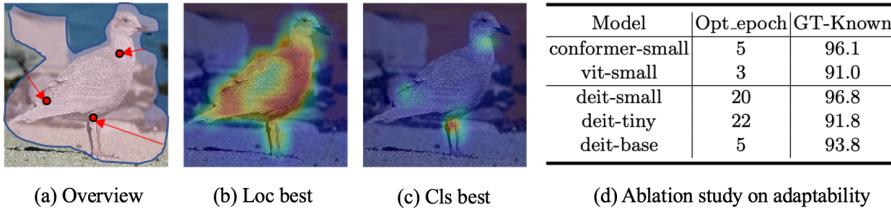


Fig. 5: (a) The overview of the activation scores propagation, which is a process that evolves from the raw attention regions to the semantic rich regions. (b) Status with the best Loc Acc at the relatively early training stage. (c) Status with the best CLS Acc at the later training stage. (d) The comparison between SCM on different Transformers and various scales. We record GT-known and the epoch number at which the best GT-known performance is obtained.

out all the experiments on Deit-small with SCM consisting of four ADBs and all the experiments share the same implementation discussed above.

Trade-off between Classification & localization. SCM is an external module and will be dropped out during inference, adding no additional computational burden. Thus there is a trade-off between performance of localization and classification when the backbone is pre-determined. As shown in Fig.5(a), SCM aims to calibrate the raw attention to localize the bird. Specifically, Transformer trained with SCM localizes objects well while suffers from sub-optimal CLS Acc in Fig.5(b). In contrast, as training process continues, it classifies objects better but only focuses on the discriminant part of the whole object, resulting in worse localization result in Fig.5(c). To clearly show the advantage of SCM for localization, we saved the model with the highest GT-Known as depicted in Fig.5(b).

Visualization Result of S^l and F^l . Implicit attention of models trained on image-level labels is blessed with remarkable localization ability as shown in CAM [39]. However, due to the effect of label-wise semantic loss, the models would finally be driven to gather around semantic-rich regions, causing the problem of partial activation. TS-CAM [9] suffers from a similar issue despite improving the localization performance by Transformer’s long-range feature dependency. In Fig.6, we display both S^l and F^l at each layer of SCM. We observe that F^0 and S^0 have already covered the object completely, demonstrating that SCM can calibrate Transformer to cover objects. As the layer gets deeper, S^l and F^l concentrate more on semantic-rich regions, and S^L at the last layer is further used to calculate the loss. It explains why we drop out SCM instead of appending it to Transformer, as sharper boundaries are provided at S^0 and F^0 .

Propagating and filtering. To understand the effect of propagating and filtering, we analyze parameters λ and β in each layer of SCM. As shown in Fig.7, the training record tells that λ in deeper layers increases, while λ in shallow layers is reduced. It indicates that SCM learns to diffuse activation at front layers while concentrating it in latter layers, verifying that SCM can enlarge partially activated regions with label-wise supervision. On the other hand, β at all layers

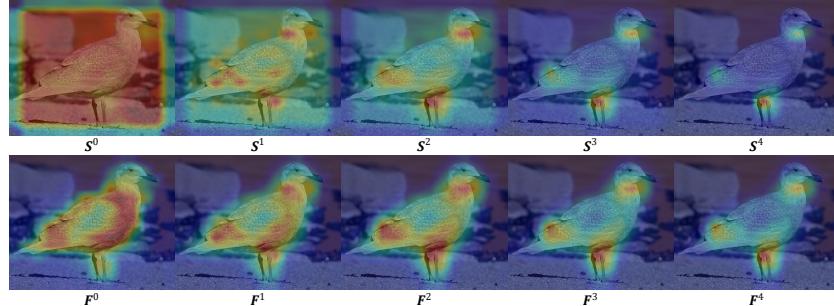


Fig. 6: Visualization of both semantic maps S^l (upper) and attention maps F^l (lower) input to the l^{th} ADB block for a sample from CUB-200-2011 test set.

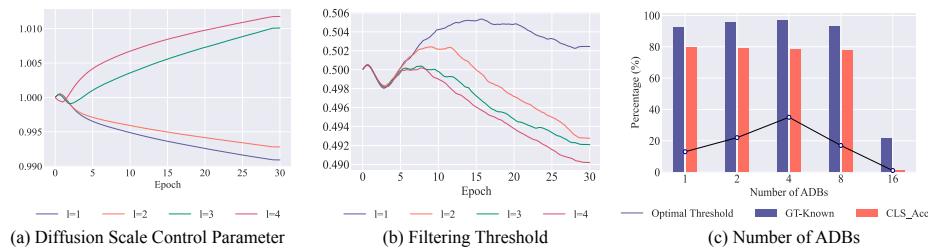


Fig. 7: The learnable parameters update when trained on Deit-small. The layer number l is shown below. (a) λ is used for the diffusion scale control, and lower λ means the wilder scale of diffusion. (b) β determines the threshold under which the activation maps should be filtered. (c) Evaluation of GT-known, Cls Acc (top-1) for different numbers of ADBs. γ (here in percentage format) denotes the threshold above which the bounding box is predicted from the score maps.

drops at the beginning, possibly because the activation provided by Transformer is sparse. It takes time for the model to shift its focus from classification to localization, as Transformer is pretrained for classification. Then it starts climbing and goes down again, indicating that attention becomes more concentrated at beginning and then turns sparse to fit the demand across layers. For instance, the front layer prefers a higher filtering threshold to reduce noise, while other layers prefer a smaller threshold to get more semantic context.

Stacking ADBs. We further investigate the effect of module scale by stacking different numbers of ADBs. As shown in Fig.7(c), we find out that the trend of GT-known and the optimal threshold almost fits the bell curve. It indicates that setting the suitable scale for SCM is essential, as when SCM becomes too deep, it fails to classify and localize objects precisely. On the other hand, the classification accuracy drops as the number of ADBs increases, while the localization performance increases first and drops later. It tells us that classification and localization are two different tasks, and we cannot obtain the optimal for both.

Adapting SCM to more situations. To evaluate SCM’s performance with other Transformers, we select ViT [27], Conformer [10] to testify SCM. Next, we compare SCM on various model scales on Deit. As shown in Fig.5(d), we record the localization performance with the optimal epoch at which the best model is saved. It turns out that SCM is successfully adapted to ViT and Conformer, which achieves satisfactory performance 91.8% and 96.1% on CUB-200-2011 respectively. On the other hand, we test SCM on Deit with different scales. Surprisingly the larger models don’t perform as well as Deit-small. It turns out that increasing model parameter size may not be optimal for SCM to obtain better performance, and the dropped optimal epoch number indicates that it may need a lower learning rate in training for better result.

Discussions. Our study presents a novel way to calibrate the Transformer for WSOL. Although we prove its adaptability to ViT [27], Conformer [10], we cannot calibrate Transformers without CLS token such as Swin [14], since CLS token is required to obtain \mathbf{F}^0 . Furthermore, it’s heuristic to choose the number of iterations used in Eqn.(6), and we simplify it as a constant number. Future research may explore methods such as Deep Reinforcement Learning to search the parameter space for the optimal diffusion policy. Furthermore, the equilibrium status Eqn.(4) is a patch-wise correlation like the self-attention matrix. It may indicate a new way to find the regions of interest by diffusion.

5 Conclusions

We proposed a simple external spatial calibration module (SCM) to refine attention and semantic representations of Vision Transformer for weakly supervised object localization (WSOL). SCM exploits the spatial and semantic coherence in images and calibrates Transformers to address the issue of partial activation. To dynamically incorporate semantic similarities and local spatial relationships of patch tokens, we propose a unified diffusion model to capture sharper object boundaries and inhibit irrelevant background activation. SCM is designed to be removed during the inference phase, and we use Transformers’ calibrated attention and semantic representations to predict localization results. Experiments on CUB-200-2011 and ILSVRC2012 datasets prove that SCM effectively covers the full objects and significantly outperforms its counterpart TS-CAM. As the first Transformer external calibration module on WSOL, we hope SCM could shed light on refining Transformers for the more challenging WSOL scenarios.

Acknowledgement. The work is supported in part by the Young Scientists Fund of the National Natural Science Foundation of China under grant No. 62106154, by Natural Science Foundation of Guangdong Province, China (General Program) under grant No.2022A1515011524, by Shenzhen Science and Technology Program ZDSYS20211021111415025, and by the Guangdong Provincial Key Laboratory of Big Data Computing, The Chinese University of Hong Kong (Shenzhen).

References

1. Bourigault, S., Lagnier, C., Lamprier, S., Denoyer, L., Gallinari, P.: Learning social network embeddings for predicting information diffusion. In: Proceedings of the 7th ACM international conference on Web search and data mining. pp. 393–402 (2014)
2. Chen, Z., Wang, C., Wang, Y., Jiang, G., Shen, Y., Tai, Y., Wang, C., Zhang, W., Cao, L.: On awakening the local continuity of transformer for weakly supervised object localization. In: Proceedings of the AAAI Conference on Artificial Intelligence (2022)
3. Cheung, G., Magli, E., Tanaka, Y., Ng, M.K.: Graph spectral image processing. Proceedings of the IEEE **106**(5), 907–930 (2018)
4. Choe, J., Oh, S.J., Lee, S., Chun, S., Akata, Z., Shim, H.: Evaluating weakly supervised object localization methods right. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3133–3142 (2020)
5. Choe, J., Shim, H.: Attention-based dropout layer for weakly supervised object localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2219–2228 (2019)
6. Choe, J., Shim, H.: Attention-based dropout layer for weakly supervised object localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2219–2228 (2019)
7. Dai, Z., Liu, H., Le, Q.V., Tan, M.: Coatnet: Marrying convolution and attention for all data sizes. Advances in Neural Information Processing Systems **34**, 3965–3977 (2021)
8. Gao, S., Tsang, I.W.H., Chia, L.T.: Laplacian sparse coding, hypergraph laplacian sparse coding, and applications. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(1), 92–104 (2013)
9. Gao, W., Wan, F., Pan, X., Peng, Z., Tian, Q., Han, Z., Zhou, B., Ye, Q.: Ts-cam: Token semantic coupled attention map for weakly supervised object localization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2886–2895 (2021)
10. Gulati, A., Qin, J., Chiu, C.C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., et al.: Conformer: Convolution-augmented transformer for speech recognition. arXiv preprint arXiv:2005.08100 (2020)
11. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015)
12. Kim, E., Kim, S., Lee, J., Kim, H., Yoon, S.: Bridging the gap between classification and localization for weakly supervised object localization. arXiv preprint arXiv:2204.00220 (2022)
13. Kondor, R.I., Lafferty, J.: Diffusion kernels on graphs and other discrete structures. In: Proceedings of the 19th international conference on machine learning. vol. 2002, pp. 315–322 (2002)
14. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. 2021 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 9992–10002 (2021)
15. Liu, Z., Li, X., Luo, P., Loy, C.C., Tang, X.: Semantic image segmentation via deep parsing network. In: Proceedings of the IEEE international conference on computer vision. pp. 1377–1385 (2015)
16. Liu, Z., Li, X., Luo, P., Loy, C.C., Tang, X.: Deep learning markov random field for semantic segmentation. IEEE transactions on pattern analysis and machine intelligence **40**(8), 1814–1828 (2017)

17. Loshchilov, I., Hutter, F.: Fixing weight decay regularization in adam (2018)
18. Lu, W., Jia, X., Xie, W., Shen, L., Zhou, Y., Duan, J.: Geometry constrained weakly supervised object localization. In: European Conference on Computer Vision. pp. 481–496. Springer (2020)
19. Ma, H., King, I., Lyu, M.R.: Mining web graphs for recommendations. IEEE Transactions on Knowledge and Data Engineering **24**(6), 1051–1064 (2011)
20. Mai, J., Yang, M., Luo, W.: 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. pp. 8766–8775 (2020)
21. Meng, M., Zhang, T., Yang, W., Zhao, J., Zhang, Y., Wu, F.: Diverse complementary part mining for weakly supervised object localization. IEEE Transactions on Image Processing **31**, 1774–1788 (2022)
22. Pan, V.: Fast and efficient parallel algorithms for the exact inversion of integer matrices. In: International Conference on Foundations of Software Technology and Theoretical Computer Science. pp. 504–521. Springer (1985)
23. Pan, V., Reif, J.: Efficient parallel solution of linear systems. In: Proceedings of the seventeenth annual ACM symposium on Theory of computing. pp. 143–152 (1985)
24. Pan, X., Ge, C., Lu, R., Song, S., Chen, G., Huang, Z., Huang, G.: On the integration of self-attention and convolution. arXiv preprint arXiv:2111.14556 (2021)
25. Qi, Y., Suhail, Y., Lin, Y.y., Boeke, J.D., Bader, J.S.: Finding friends and enemies in an enemies-only network: a graph diffusion kernel for predicting novel genetic interactions and co-complex membership from yeast genetic interactions. Genome research **18**(12), 1991–2004 (2008)
26. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (2015)
27. Sharir, G., Noy, A., Zelnik-Manor, L.: An image is worth 16x16 words, what is a video worth? arXiv preprint arXiv:2103.13915 (2021)
28. Singh, K.K., Lee, Y.J.: Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization (2017)
29. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning. pp. 10347–10357. PMLR (2021)
30. Wei, J., Wang, Q., Li, Z., Wang, S., Zhou, S.K., Cui, S.: Shallow feature matters for weakly supervised object localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5993–6001 (2021)
31. Wei, J., Wang, S., Zhou, S.K., Cui, S., Li, Z.: Weakly supervised object localization through inter-class feature similarity and intra-class appearance consistency. In: European conference on Computer Vision. Springer (2022)
32. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-ucsd birds 200. Tech. rep. (2010)
33. Xue, H., Liu, C., Wan, F., Jiao, J., Ji, X., Ye, Q.: Danet: Divergent activation for weakly supervised object localization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6589–6598 (2019)
34. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6023–6032 (2019)
35. Zhang, C.L., Cao, Y.H., Wu, J.: Rethinking the route towards weakly supervised object localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13460–13469 (2020)

36. Zhang, X., Wei, Y., Feng, J., Yang, Y., Huang, T.S.: Adversarial complementary learning for weakly supervised object localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1325–1334 (2018)
37. Zhang, X., Wei, Y., Kang, G., Yang, Y., Huang, T.: Self-produced guidance for weakly-supervised object localization. In: Proceedings of the European conference on computer vision (ECCV). pp. 597–613 (2018)
38. Zhang, X., Wei, Y., Yang, Y.: Inter-image communication for weakly supervised localization. In: European Conference on Computer Vision. pp. 271–287. Springer (2020)
39. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2921–2929 (2016)

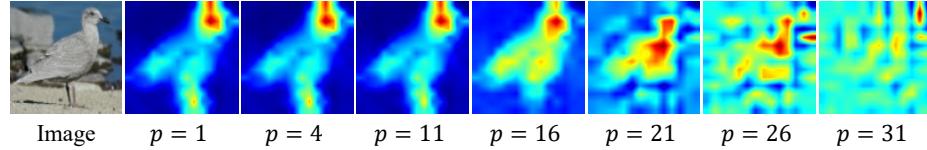


Fig. 8: Illustration of approximation by Newton Schulz Iteration. Each map denotes the redistributed \mathbf{F} with corresponding number of iterations p below. In the main paper, we use iteration number $p = 4$.

A Additional ablation study

We give more experimental results and analysis on Spatial Calibration Module (SCM) proposed in the main paper. Firstly, we conduct more ablation studies on the activation diffusion module, especially on the Newton Schulz Approximation iteration. Next, we study various strategies of combining \mathbf{S}^l and \mathbf{F}^l and testify its influence on localization. To test SCM on more challenging measures, we validate it on MaxboxAcc. Furthermore, we provide the complete proof of the semantics-coupled Laplacian matrix \mathbf{L}^l at Eqn.(4) in the main paper, followed by a theoretical analysis of the semantic flow redistribution.

In this section, we conduct experiments on the influence of the iteration number in Newton Schulz Iteration. We also test the methodology to build up the final prediction score map using maps from various layers.

A.1 Selecting different iteration numbers

As shown in Fig.8, we observe that the approximation of \mathbf{L}^{-1} by Newton Schulz can be accelerated with the increasing number of iterations. It raises the question of what its impact on the localization performance is. To answer this question, we train several models with four ADB layers following the same setting as the main paper, except that the number of iterations varies.

As depicted in Fig.10, we plot GT-Known and the hyperparameter threshold γ above which we generate the binary map. It turns out that iteration $p = 4$ is still the optimal choice that exceeds other settings over 5% in GT-Known. To explore the reason, we plot γ and observe that the iteration $p = 4$ yields a much larger region of interest than others. It indicates that SCM may need a relatively small number of iterations in each block, or semantic information would be over-diffused, resulting in degraded performance.

A.2 Strategies on combining maps

We design SCM as an external module that calibrates Transformer trained on the classification to weakly supervised localization scenarios. During inference, SCM will be dropped out, so we only use \mathbf{S}^0 and \mathbf{F}^0 for prediction. We further explore whether combining maps from other blocks would yield a different result.

As shown in Fig.9, we produce the activation by combining \mathbf{S}^l and \mathbf{F}^l and depict it in a pair-wise way. We find out that \mathbf{S}^l tends to concentrate more on

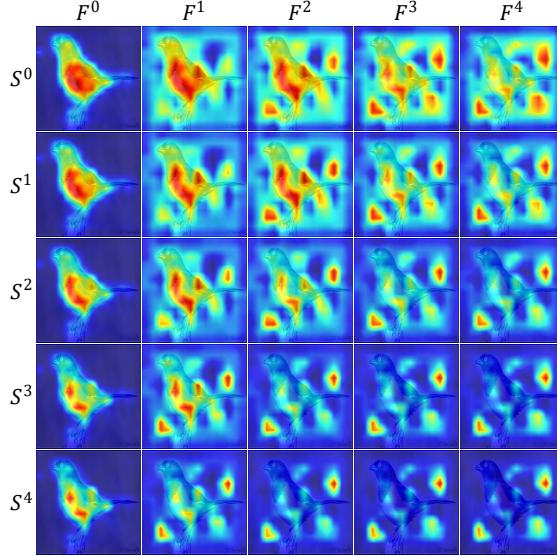


Fig. 9: Illustration of coupling semantic maps and attention maps across layers. Sources of each image are indicated at corresponding row and column.

semantic-rich regions as the number of layers increases. On the other hand, \mathbf{F}^l shows a similar pattern as the layer goes deeper. The reason is that the semantic token maps \mathbf{S}^l are supervised by the label loss that drives the model to focus on discriminative parts. However, different from the naive transformer implementation (TS-CAM), the Transformer with SCM learns to calibrate semantic and attention maps through backpropagation, as we can observe that it revises the coupled activation with more spatial details and clear boundaries in upper layers. At last, the refined coupled score map \mathbf{S}^0 and \mathbf{F}^0 becomes a promising candidate for localization.

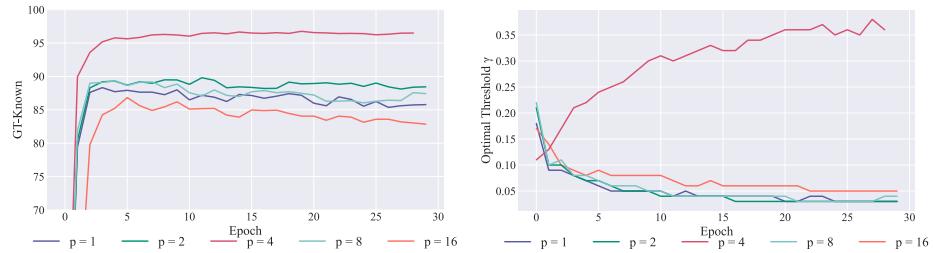


Fig. 10: Illustration of the GT-Known performance and the optimal filtering threshold γ with various number of Newton Schulz Iterations p in validation. γ determines the threshold above which the bounding box is predicted from the score maps, which means γ is proportional to the activated region.

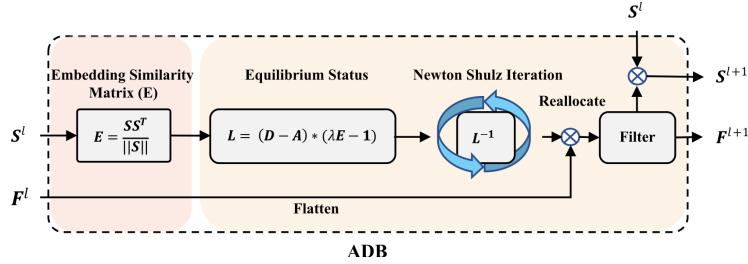


Fig. 11: Architecture of $(l+1)^{th}$ Activation Diffusion Block (ADB).

A.3 Evaluation result on other metrics

MaxboxAcc is reformulated to further GT-Known (the same fixed δ 50%) with the optimal threshold γ in generating the binary map. Compared with GT-Known, MaxboxAcc precludes misleading hyperparameter γ that depends heavily on the data and model architecture. MaxboxAccV2 with optimal γ , is a more strict measure than the former MaxboxAccV1. (1) It averages the performance across $\delta \in \{0.3, 0.5, 0.7\}$ to address diverse demand for localization fitness. (2) It considers the best match between the set of all estimated boxes and the set of all ground-truth boxes as prediction, instead of only one box prediction from the largest connected component of the score map in prior methods.

Towards a well-posed setup on WSOL, which is trained without any localization supervision, we shift the evaluation on a held-out set CUBV2 [4] not overlapping with the available validation set (now the test set). Then we evaluate both SCM and TS-CAM on it with the metrics MaxboxAccV1 and MaxboxAccV2 in the experiment shown in Table.2 for the reason that selecting hyperparameter γ with full supervision in the test set violates the principle of WSOL. To make a fair comparison with the evaluation results given in [4], we keep the same training budget with fixed training epochs to 50 and a fixed batch size of 32 and save the models, including SCM and TS-CAM with the best MaxboxAccV1 or MaxboxAccV2 on CUBV2.

In Table.2, we compare TS-CAM and SCM on CUBV2 [4] on the same computational budget as previous methods. It turns out that both TS-CAM and SCM have achieved satisfactory performance, but SCM surpasses TS-CAM by 7.7% and 10.3% on MaxboxAccV1 and MaxboxAccV2, respectively. Furthermore, a higher MaxboxAccV2 score proves that SCM has great adaptability and attends to various levels of localization fitness demands.

B More details on activation diffusion

Over the past decades, Transformer has had tremendous success, largely attributed to its efficient attention mechanism to capture the long-range dependency. However, its limitations cannot be ignored. Studies have found that the transformer has a natural limitation on local context modeling [7,24], which is

Table 2: Comparison of SCM by MaxboxAcc [4] on CUB [32]. Values in bracket shows improvement of our method compared with TS-CAM [9].

Model	Backbone	MaxboxAccV1	MaxboxAccV2
CAM[39]	VGG16	71.1	63.7
ACoL[36]	VGG16	72.3	57.4
ADL[6]	VGG16	75.7	66.3
CutMix[34]	VGG16	71.9	62.3
SPG[37]	InceptionV3	62.7	55.9
ADL[6]	InceptionV3	63.4	58.8
PDM[21]	Resnet50	-	70.7
BGC[12]	Resnet50	-	80.1
TS-CAM[9]	Deit-S	88.9	79.6
SCM(ours)	Deit-S	96.6 (7.7↑)	89.9 (10.3↑)

* The experiment is iteratively trained one epoch on CUB train set and evaluated one epoch on CUBV2 [4]. The annotation mapping in the counterpart ISLVRCV2 [4] is currently not available, so we evaluate TS-CAM and SCM only on CUB-200-2011.

critical for the object localization task. We further extend its ability by introducing SCM that calibrates the Transformer to embrace spatial and semantic coherence to solve this issue.

As shown in Fig.11, we apply Activation Diffusion Block in SCM to reallocate the activation region \mathbf{F} . Here, we give the detailed stpdf to get the Laplacian matrix \mathbf{L} which denotes "equilibrium status" at Eqn.(4) in our main paper.

This section will describe the activation diffusion behind a physic evolution model on a network structure in detail. We start with an introduction to the diffusion process that enables the exchange of information among vertexes. Next, we further analyze diffusion behavior with semantics on a global scale. At last, we show how to get the re-allocated attention map. Firstly, we build a graph $G \langle V, E \rangle$, where V and E represent the set of vertexes and edges, respectively. Also, \mathbf{v}_i denotes a vertex in V , and $e_{i,j}$ in E denotes an edge between \mathbf{v}_i and \mathbf{v}_j , and we define the information flow as $\mathbf{I} \in \mathbb{R}^N$, where N is the number of patches. G is shown in Fig.12, where we display the flow exchange between \mathbf{v}_i and its first-order neighbors $\mathbf{v}_{i-1}, \mathbf{v}_{i+1}, \mathbf{v}_{i-W}, \mathbf{v}_{i+W}$, where (H, W) is the 2D patch resolution and we use the token sequence indexes to denote the spatially connected four neighbors.

To make diffusion semantic-aware in G , as shown in Fig.12, we design a model to describe both the flow influx and the outflux on \mathbf{v}_i . Firstly, the flow input is based on the initial activation maps, where the activation score is proportional to the input rate; another source is the neighbor nodes as \mathbf{v}_i will share flow with them. On the other hand, the flow will go outwards to nearby ones simultaneously, and to make it semantics-aware, we introduce the 'semantic flow' that escapes from the nodes. Thus, The rate of fluid change in \mathbf{v}_i at time t could be

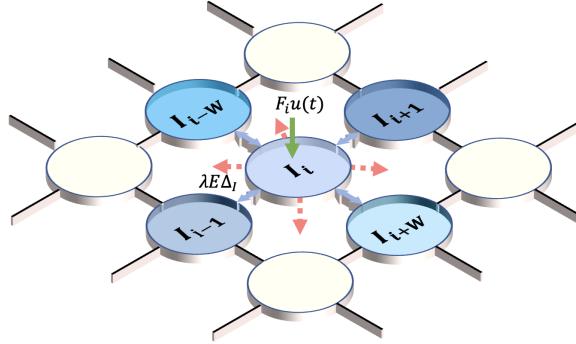


Fig. 12: Illustration of diffusion for v_i and its first-order neighbors, where (H, W) is the reshaped 2D graph resolution, where H denotes the number of nodes per column, and W denotes the number of nodes per row. Each circle represents a patch in this graph, and we denote the patch sequence indexes on top of them. The arrows represent flow change with the horizontal direction that denotes exchange with neighbor vertexes, and the vertical represents input and output for G . We further specify types of exchange by different colors, where (Green) $F_i u(t)$ is the initial input rate; (Blue) The communication rate with neighbors; (Red) The rate of semantic flow which is related to the embedding similarity and the amount of flow.

described as,

$$\hat{\mathbf{I}}_i(t) = \underbrace{(\mathbf{F}_i u(t) + \sum_j \mathbf{A}_{i,j} \mathbf{I}_i(t))}_{influx} - \underbrace{(\sum_j \mathbf{A}_{j,i} \mathbf{I}_j(t) + \lambda \sum_j \mathbf{A}_{i,j} (\mathbf{I}_i(t) - \mathbf{I}_j(t)) \mathbf{E}_{i,j})}_{outflux} \quad (8)$$

where λ is a learnable parameter for flexible control over the scale of diffusion. Specifically, for each v_i , the input for G exists if v_i is one of the source nodes, then the input rate is $\mathbf{F}_i u(t)$, i.e. the score maps $\mathbf{F}_i > 0$ then v_i can be treated as the source. Next the input from the direct neighbors should also be considered, given $\sum_j \mathbf{A}_{i,j} \mathbf{I}_i(t)$. On the other hand, for output, when propagating from v_i to v_j , there exists the semantic flow which penalizes the flow exchange with low semantic similarity, i.e. the cosine distance $\mathbf{E}_{i,j}$. Thus, $\lambda \sum_j \mathbf{A}_{i,j} (\mathbf{I}_i(t) - \mathbf{I}_j(t)) \mathbf{E}_{i,j}$ describes the escaped semantic flows for the propagation from v_i to its neighbor, denoted as red arrows in Fig.12. λ is a hyperparameters to adjust the overall contribution of semantic flow. Next, the outflux into the direct neighbors is $\sum_j \mathbf{A}_{j,i} \mathbf{I}_j(t)$.

Then we can study the dynamic change of flow regarding $G \langle V, E \rangle$ and describe the graph's response to the flow dynamics. Eqn.(8) could be further extended to the global scale,

$$\hat{\mathbf{I}}(t) = \mathbf{L}\mathbf{I}(t) + u(t)\Gamma(\mathbf{F}) \quad (9)$$

where

$$\mathbf{L} = (\mathbf{D} - \mathbf{A}) * (1 - \lambda \mathbf{E}) \quad (10)$$

Eqn.(10) is the shifted Laplacian matrix and Γ is a flatten operator used to reshape \mathbf{F} into a sequence.

Eqn.(8) tells the flow at \mathbf{v}_i that changes with time. Next, we could further take the integral to accumulate the total changes within a certain amount of time, which could be used to describe the trend of flow at G . Thus, from Eqn.(9), we obtain the expression of the amount of flow in G by,

$$\mathbf{I}(t) = \int_{t'=0}^t e^{-\mathbf{L}(t-t')\Gamma(\mathbf{F})u(t')} dt' \quad (11)$$

Eqn.(11) tells us that the graph is dynamically adjusted by semantic embedding similarity \mathbf{E} with spatial relationship. Denote a special time t_0 when $\hat{\mathbf{I}}(t_0) = 0$, we consider the 'equilibrium' status is reached as the influx rate equals the outflux rate for \mathbf{v}_i . As $t_0 \in [0, \infty]$, when $t \rightarrow \infty$, the total amount of flow in G will not change and we obtain,

$$\lim_{t \rightarrow \infty} \mathbf{I}(t) = \mathbf{L}^{-1} \Gamma(\mathbf{F}) \quad (12)$$

Eqn.(12) implies the fully-diffused activation, however, as discussed in our main paper, \mathbf{L} is not guaranteed to be positive-definite, and its inverse may not exist. Meanwhile, as observed in our initial experiments in Fig.8, directly applying the inverse has produced unwanted artifacts that may downgrade localization quality. Thanks to the Newton Schulz method, we exploit its great convergence ability that approximates \mathbf{L}^{-1} with a few numbers of iterations. As shown in Fig.11, we couple the approximated \mathbf{L}^{-1} to incorporate spatial and semantic correlation into \mathbf{F} in the end, which is shown in Eqn.(12).