Supplementary Material

Haotian Bai*, Ruimao Zhang **, Jiong Wang, and Xiang Wan

Shenzhen Research Institute of Big Data, The Chinese Univeristy of Hong Kong (Shenzhen), China
haotianwhite@outlook.com, zhangruimao@cuhk.edu.cn

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 combing $\boldsymbol{S}^l$ and $\boldsymbol{F}^l$ 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 $\boldsymbol{L}^l$ at Eqn.(4) in the main paper, followed by a theoretical analysis of the semantic flow redistribution.

1 Additional ablation study

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.

1.1 Selecting different iteration numbers

As shown in Fig.1, we observe that the approximation of $\boldsymbol{L}^{-1}$ 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.3, 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.

1.2 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,

* Research done when Haotian Bai was a Research Assistant at The Chinese University of Hong Kong, shenzhen.

** Corresponding Author

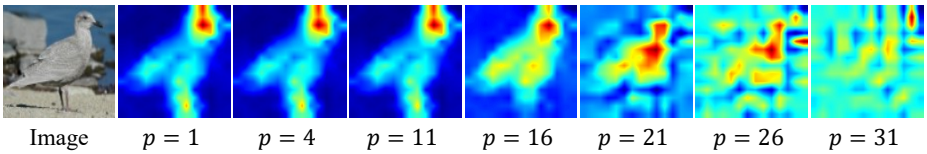

Fig. 1: Illustration of approximation by Newton Schulz Iteration. Each map denotes the redistributed $\boldsymbol{F}$ with corresponding number of iterations p below. In the main paper, we use iteration number $p = 4$.

SCM will be dropped out, so we only use $\boldsymbol{S}^0$ and $\boldsymbol{F}^0$ for prediction. We further explore whether combing maps from other blocks would yield a different result.

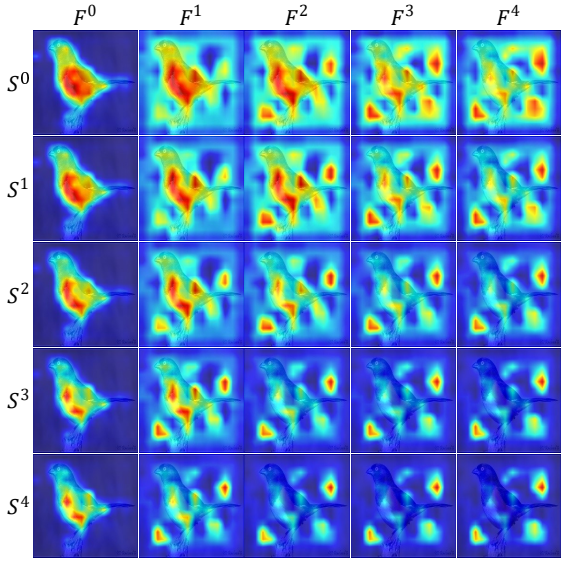

Fig. 2: Illustration of coupling semantic maps and attention maps across layers. Sources of each image are indicated at corresponding row and column.

As shown in Fig.2, we produce the activation by combining $\boldsymbol{S}^l$ and $\boldsymbol{F}^l$ and depict it in a pair-wise way. We find out that $\boldsymbol{S}^l$ tends to concentrate more on semantic-rich regions as the number of layers increases. On the other hand, $\boldsymbol{F}^l$ shows a similar pattern as the layer goes deeper. The reason is that the semantic token maps $\boldsymbol{S}^l$ 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 $\boldsymbol{S}^0$ and $\boldsymbol{F}^0$ becomes a promising candidate for localization.

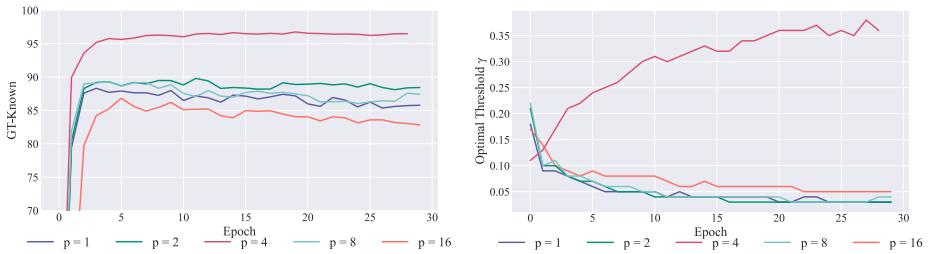

Fig. 3: Illustration of the GT-Known performance and the optimal filtering threshold γ with various number of Newton Schulz Iterations p in validation. γ determines the threshold above which the bounding box is predicted from the score maps, which means γ is proportional to the activated region.

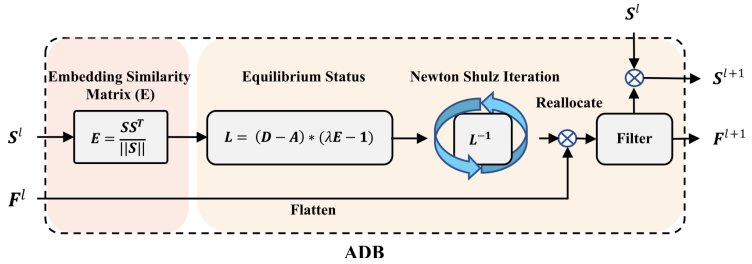

Fig. 4: Architecture of $(l+1)^{th}$ Activation Diffusion Block (ADB).

1.3 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 [1] not overlap-

Table 1: Comparison of SCM by MaxboxAcc [1] on CUB [8]. Values in bracket shows improvement of our method compared with TS-CAM [4].

Model	Backbone	MaxboxAccV1	MaxboxAccV2
CAM[12]	VGG16	71.1	63.7
ACoL[10]	VGG16	72.3	57.4
ADL[2]	VGG16	75.7	66.3
CutMix[9]	VGG16	71.9	62.3
SPG[11]	InceptionV3	62.7	55.9
ADL[2]	InceptionV3	63.4	58.8
PDM[6]	Resnet50	-	70.7
BGC[5]	Resnet50	-	80.1
TS-CAM[4]	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 [1]. The annotation mapping in the counterpart ISLVRCV2 [1] is currently not available, so we evaluate TS-CAM and SCM only on CUB-200-2011.

ping 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.1 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 [1], 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.1, we compare TS-CAM and SCM on CUBV2 [1] 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.

2 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 [3, 7], which is 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.4, we apply Activation Diffusion Block in SCM to reallocate the activation region $\boldsymbol{F}$. Here, we give the detailed steps to get the Laplacian matrix $\boldsymbol{L}$ which denotes "equilibrium status" at Eqn.(4) in our main paper.

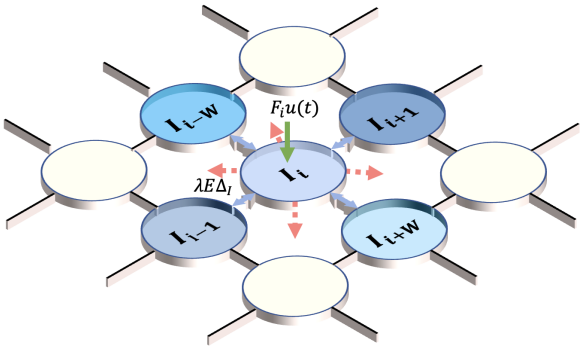

Fig. 5: Illustration of diffusion for $\boldsymbol{v}_i$ and its first-order neighbors, where (H, W) is the reshaped 2D graph resolution, where H denotes the number of nodes per column, and W denotes the number of nodes per row. Each circle represents a patch in this graph, and we denote the patch sequence 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) $\boldsymbol{F}_i u(t)$ 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.

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, $\boldsymbol{v}_i$ denotes a vertex in V, and $e_{i,j}$ in E denotes an edge between $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$, and we define the information flow as $\boldsymbol{I} \in \mathbb{R}^N$, where N is the number of patches. G is shown in Fig.5, where we display the flow exchange between $\boldsymbol{v}_i$ and its first-order neighbors $\boldsymbol{v}_{i-1}$, $\boldsymbol{v}_{i+1}$, $\boldsymbol{v}_{i-W}$, $\boldsymbol{v}_{i+W}$, 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.5, we design a model to describe both the flow influx and the outflux on $\boldsymbol{v}_i$. 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 $\boldsymbol{v}_i$ 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 $\boldsymbol{v}_i$ at time t could be

described as,

$$\hat{\boldsymbol{I}_i}(t) = \underbrace{(\boldsymbol{F}_i u(t) + \sum_j \boldsymbol{A}_{i,j}\boldsymbol{I}_i(t))}_{influx} - \underbrace{(\sum_j \boldsymbol{A}_{j,i}\boldsymbol{I}_j(t) + \lambda \sum_j \boldsymbol{A}_{i,j}(\boldsymbol{I}_i(t) - \boldsymbol{I}_j(t))\boldsymbol{E}_{i,j})}_{outflux} \quad (1)$$

where λ is a learnable parameter for flexible control over the scale of diffusion. Specifically, for each $\boldsymbol{v}_i$, the input for G exists if $\boldsymbol{v}_i$ is one of the source nodes, then the input rate is $\boldsymbol{F}_i u(t)$, *i.e.* the score maps $\boldsymbol{F}_i > 0$ then $\boldsymbol{v}_i$ can be treated as the source. Next the input from the direct neighbors should also be considered, given $\sum_j \boldsymbol{A}_{i,j}\boldsymbol{I}_i(t)$. On the other hand, for output, when propagating from $\boldsymbol{v}_i$ to $\boldsymbol{v}_j$, there exists the semantic flow which penalizes the flow exchange with low semantic similarity, *i.e.* the cosine distance $\boldsymbol{E}_{i,j}$. Thus, $\lambda \sum_j \boldsymbol{A}_{i,j}(\boldsymbol{I}_i(t) - \boldsymbol{I}_j(t))\boldsymbol{E}_{i,j})$ describes the escaped semantic flows for the propagation from $\boldsymbol{v}_i$ to its neighbor, denoted as red arrows in Fig.5. λ is a hyperparameters to adjust the overall contribution of semantic flow. Next, the outflux into the direct neighbors is $\sum_j \boldsymbol{A}_{j,i}\boldsymbol{I}_j(t)$.

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.(1) could be further extended to the global scale,

$$\hat{\boldsymbol{I}}(t) = \boldsymbol{L}\boldsymbol{I}(t) + u(t)\Gamma(\boldsymbol{F}) \quad (2)$$

where

$$\boldsymbol{L} = (\boldsymbol{D} - \boldsymbol{A}) * (1 - \lambda \boldsymbol{E}) \quad (3)$$

Eqn.(3) is the shifted Laplacian matrix and Γ is a flatten operator used to reshape $\boldsymbol{F}$ into a sequence.

Eqn.(1) tells the flow at $\boldsymbol{v}_i$ 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.(2), we obtain the expression of the amount of flow in G by,

$$\boldsymbol{I}(t) = \int_{t'=0}^{t} e^{-\boldsymbol{L}(t-t')\Gamma(\boldsymbol{F})u(t')dt'} \quad (4)$$

Eqn.(4) tells us that the graph is dynamically adjusted by semantic embedding similarity $\boldsymbol{E}$ with spatial relationship. Denote a special time t_0 when $\hat{\boldsymbol{I}(t_0)} = 0$, we consider the 'equilibrium' status is reached as the influx rate equals the outflux rate for $\boldsymbol{v}_i$. As $t_0 \in [0, \infty]$, when $t \to \infty$, the total amount of flow in G will not change and we obtain,

$$\lim_{t\to\infty} \boldsymbol{I}(t) = \boldsymbol{L}^{-1}\Gamma(\boldsymbol{F}) \quad (5)$$

Eqn.(5) implies the fully-diffused activation, however, as discussed in our main paper, $\boldsymbol{L}$ is not guaranteed to be positive-definite, and its inverse may

not exist. Meanwhile, as observed in our initial experiments in Fig.1, 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 $\boldsymbol{L}^{-1}$ with a few numbers of iterations. As shown in Fig.4, we couple the approximated $\boldsymbol{L}^{-1}$ to incorporate spatial and semantic correlation into $\boldsymbol{F}$ in the end, which is shown in Eqn.(5).

References

1. 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)
2. 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)
3. 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)
4. 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)
5. 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)
6. 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)
7. 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)
8. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-ucsd birds 200. Tech. rep. (2010)
9. 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)
10. 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)
11. 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)
12. 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)