# Appendix

Anonymous CVPR 2022 submission

Paper ID 1744

## 1. Supplementary of Method

### 1.1. Key symbols List

Here we give the pivotal symbols list in Table. 1. In general, we use ***italic bold*** uppercase characters to denote the matrices. Vectors are denoted with lowercase. Sets are noted by blackboard bold, for example $\mathbb{X}$.

Table 1. The Meaning of Some Pivotal Symbols

| | |
|---|---|
| $\boldsymbol{X} \in \mathbb{R}^{3\times N}$ | RGB feature of the input image. |
| $\boldsymbol{Z} \in \mathbb{R}^{C\times N}$ | The pixel-level feature of $\boldsymbol{X}$ |
| $\boldsymbol{z} \in \mathbb{R}^{C\times 1}$ | Image-level feature of $\boldsymbol{X}$. |
| $\boldsymbol{Y} \in \mathbb{R}^{K\times N}$ | The pixel-level ground truth mask. |
| $\boldsymbol{y} \in \mathbb{R}^{K\times 1}$ | The image-level ground truth mask. |
| $\boldsymbol{Y}^* \in \mathbb{R}^{K\times N}$ | The predict localization score. |
| $\boldsymbol{y}^* \in \mathbb{R}^{K\times 1}$ | The predict classification score. |
| $\boldsymbol{s} \in \mathbb{R}^{C\times 1}$ | Feature of sample in source domain. |
| $\boldsymbol{t} \in \mathbb{R}^{C\times 1}$ | Feature of sample in target domain. |
| $\boldsymbol{M} \in \mathbb{R}^{K+1\times C}$ | Cache matrix of the proposed TSA. |
| $\boldsymbol{r} \in \mathbb{R}^{K+1\times 1}$ | Matrix that contains update ratio. |
| $\boldsymbol{a}^t \in \mathbb{R}^{C\times 1}$ | Anchor of real target domain. |
| $\boldsymbol{a}^u \in \mathbb{R}^{C\times 1}$ | Anchor of Universum target domain. |
| $\boldsymbol{C}^{init} \in \mathbb{R}^{3\times 1}$ | Initial cluster center of K-Means. |
| $\boldsymbol{C} \in \mathbb{R}^{3\times 1}$ | Cluster center outputted by K-Means. |
| $\mathbb{X} : \{\boldsymbol{X}\}$ | Training image set. |
| $\mathbb{S} : \{\boldsymbol{s}\}$ | Source domain/set. |
| $\mathbb{T} : \{\boldsymbol{t}\}$ | Target domain/set. |
| $\mathbb{Y}^s : \{\boldsymbol{y}^s = \boldsymbol{y}\}$ | Label set for source domain. |
| $\mathbb{Y}^t : \{\boldsymbol{y}^t = \boldsymbol{Y}_{:,i}\}$ | Label set for target domain. |
| $\mathbb{T}^f : \{\boldsymbol{t}^f\}$ | Fake Target domain/set. |
| $\mathbb{T}^t : \{\boldsymbol{t}^t\}$ | Real Target domain/set. |
| $\mathbb{T}^u : \{\boldsymbol{t}^u\}$ | Universum Target domain/set. |
| $N$ | The number of pixels (height*width). |
| $C$ | The number of channel for feature. |
| $K$ | The number of object class. |
| $M$ | The number of images in training set. |
| $y^c$ | The cluster label outputted by K-Means. |
| $f(\cdot) : \mathbb{R}^{3\times N} \rightarrow \mathbb{R}^{C\times N}$ | The feature extractor. |
| $g(\cdot) : \mathbb{R}^{?\times N} \rightarrow \mathbb{R}^{?\times 1}$ | The feature aggregator. |
| $e(\cdot) : \mathbb{R}^{C\times ?} \rightarrow \mathbb{R}^{K\times ?}$ | The score estimator. |
| $\mathcal{L}(\mathbb{S}, \mathbb{Y}^s, \mathbb{T})$ | The entire loss function. |
| $\mathcal{L}_c(\mathbb{S}, \mathbb{Y}^s)$ | The classification loss. |
| $\mathcal{L}_d(\mathbb{S} \cup \mathbb{T}^f, \mathbb{T}^t)$ | The domain adaption loss of DAL. |
| $\mathcal{L}_u(\mathbb{T}^u)$ | The Universum regularization of DAL. |

### 1.2. Equivalency between CAM and MIL

Most WSOL method follows the CAM that utilizes the classification structure for object localization. Here we show that the mechanism of the CAM-based methods is equal to the classification under MIL manner.

*Proof.* In CAM-based methods, the pixel-level feature map $\boldsymbol{Z} \in \mathbb{R}^{C\times N}$ is fed into the GAP-based aggregator to generate the image feature $\boldsymbol{z} \in \mathbb{R}^{C\times 1}$. Then the estimator with learning weight $\mathbf{W} \in \mathbb{R}^{K\times C}$ is operated on $\boldsymbol{z}$ to generate the image-level classification scores. Assuming that the classification score is $\boldsymbol{y}^* \in \mathbb{R}^{K\times 1}$ and localization score is $\boldsymbol{Y}^* \in \mathbb{R}^{K\times N}$, the workflow of CAM can be reformulated as:

$$\begin{aligned} \boldsymbol{y}^* &= \mathbf{W} * \boldsymbol{z} = \sum_c^C \mathbf{W}_{:,c}\boldsymbol{z}_c = \frac{1}{N}\sum_c^C \mathbf{W}_{:,c} * (\sum_n^N \boldsymbol{Z}_{c,n})) \\ &= \frac{1}{N}\sum_c^C\sum_n^N \mathbf{W}_{:,c} * \boldsymbol{Z}_{c,n} = \frac{1}{N}\sum_n^N(\sum_c^C \mathbf{W}_{:,c} * \boldsymbol{Z}_{c,n}) \\ &= \frac{1}{N}\sum_n^N(\mathbf{W} * \boldsymbol{Z}_{:,n}) = \frac{1}{N}\sum_n^N \mathbf{Y}^*_{:,n} \end{aligned} \tag{1}$$

Eq. 1 shows that CAM is equal to firstly generating the classification score of the instance (pixel/patch) $\mathbf{Y}^*_{:,n}$, and then determining the classification score of the bag (image) $\boldsymbol{y}^*$ based on the mean of $\mathbf{Y}^*_{:,n}$. Thus, it is the same as doing image classification under the MIL manner, where pixel/patches are the instance and images are the bag. □

### 1.3. Feature-based Universum Regularization

The Universum regularization proposed by Jason [10] directly adopt $l_1$ regularization on the classification score of Universum samples to promote the learned classifier. Here we prove that minimizing our proposed $\mathcal{L}_u(\mathbb{T}^u)$ is equal to minimizing the upper bound of it.

*Proof.* By defining the weight matrix of the estimator/classifier as $\mathbf{W} \in \mathbb{R}^{K\times C}$, we can reformulate the original classification-based Universum regularization:

$$
\begin{aligned}
\mathcal{L}_u^c &= \sum_{t^u \in \mathbb{T}^u} |\mathbf{W} * \boldsymbol{t}^u| \\
&= \sum_{t^u \in \mathbb{T}^u} \sum_{k}^{K} |\mathbf{W}_{k,:} * \boldsymbol{t}^u| \\
&= \sum_{t^u \in \mathbb{T}^u} \sum_{k}^{K} |\sum_{c}^{C} (\mathbf{W}_{k,c} * \boldsymbol{t}_c^u)| \\
&\leq \sum_{t^u \in \mathbb{T}^u} \sum_{k}^{K} |\sum_{c}^{C} \frac{(\mathbf{W}_{k,c})^2 + (\boldsymbol{t}_c^u)^2}{2}| \\
&= \frac{1}{2} \sum_{k}^{K} \sum_{c}^{C} (\mathbf{W}_{k,c})^2 + \frac{1}{2} \sum_{t^u \in \mathbb{T}^u} \sum_{c}^{C} (\boldsymbol{t}_c^u)^2
\end{aligned}
\tag{2}
$$

It can be seen that the first term, *i.e.* $\frac{1}{2}\sum_k^K \sum_c^C (\mathbf{W}_{k,c})^2$, regulars the weights of classifier, which is uncorrelated with the Universum set $\mathbb{T}^u$. Thus, we can minimize the upper bound of $\mathcal{L}_u^c$ directly by minimizing the second term of Eq. 2, which takes the same effect of the $\mathcal{L}_u(\mathbb{T}^u)$ defined in our paper. □

### 1.4. Structure for DA-WSOL with DANN

Except for the MMD [4] that does not require any additional module for domain adaption, our paper also engages the adversarial training based DANN [3] as the UDA method of our proposed DA-WSOL pipeline. The corresponding structure is shown in Fig. 1.

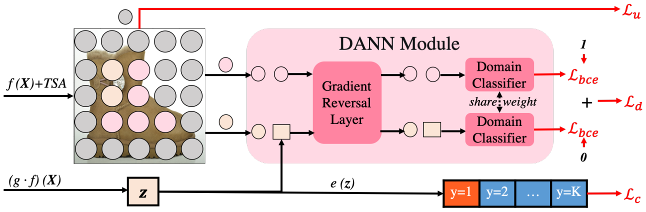

Figure 1. The structure of DA-WSOL with adopting DANN.

In detail, a domain classifier (implemented by fully-connected layer) is added to generate the domain label $y^d$ of the feature, *i.e.* discerning whether the sample belongs to the source domain ($y^d = 0$) or target domain ($y^d = 1$). Thus, the domain adaption loss $\mathcal{L}_d$ of our DA-WSOL can be implemented as:

$$
\mathcal{L}_d(\mathbb{S} \cup \mathbb{T}^f, \mathbb{T}^t) = \mathcal{L}_{bce}(\mathbb{S} \cup \mathbb{T}^f, \mathbf{0}) + \mathcal{L}_{bce}(\mathbb{T}^t, \mathbf{1}) \tag{3}
$$

Moreover, a gradient reversal layer (GRL) [3] is also added right before the domain classifier. It reverses the backward effect of the domain classifier when calculating the gradient of parameters that are upstream the GRL (parameters of feature estimator and feature aggregator). With the help of GRL, the parameters of the domain classifier are updated to minimize $\mathcal{L}_d$, while the parameters of the feature extractor and aggregator are updated to maximize $\mathcal{L}_d$, *i.e.* learning domain-invariable features.

## 2. Additional Experiments

### 2.1. Interpretability of the target sampling strategy

We verify the interpretability of our proposed target sampling strategy, which selects the representative samples for different types of the target domain. Fig. 2 visualizes the sampling results of the three target sets. It shows the fake target set $\mathbb{T}^f$ tends to catch the target samples (pixels) that generally exist in most images, such as head/wing of the cock, head of the lion, screen of the cell phone. This makes their features more discriminative and causes their high importance when aggregating the source feature by $g(\cdot)$. Thus, the features of these target samples are more similar to the source features and can be used to supply the insufficient samples of the source domain. While, the real target $\mathbb{T}^t$ tends to catch target samples that are less discriminative than the samples of $\mathbb{T}^f$. These target samples can be seen as the hard samples that have large feature discrepancy with the source domain. Aligning the feature distribution between these hard samples and source samples can efficiently enhance the estimator to identify these less discriminating object locations. Moreover, the Universum target set $\mathbb{T}^u$ also effectively catches the Universum sample, *i.e.* background locations, whose label is unseen in the source domain (because the background does not belong to the class of any object). Thus, $\mathbb{T}^u$ helps to purify the background locations from $\mathbb{T}^t$ and $\mathbb{T}^r$. These three subsets help to better solve the domain adaption in the WSOL scenario and contribute to the high performance of our DA-WSOL.

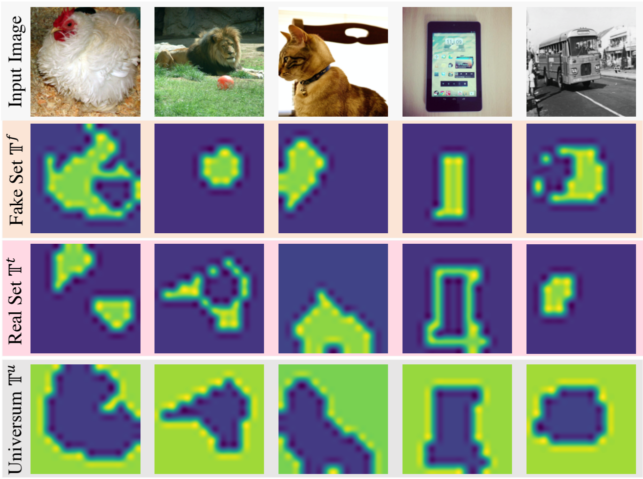

Figure 2. The visualization of the assigning results, where masks are collected based on the cluster label $y^c = 0, 1, 2$ respectively for $\mathbb{T}^u$, $\mathbb{T}^t$, $\mathbb{T}^r$ as indicated in Eq.(6) of our paper.

## 2.2. Influence of the number of selecting samples

Considering the number of selecting samples of three target subsets are determined by the hyper-parameter $n$, here we test the influence of this hyper-parameter for the localization performance. Fig. 3 shows selecting different number samples for three subsets. It can be seen that setting $n = 32$ is the most effective setting, because if $n$ is too small, the lack number of samples cannot effectively estimate the target distribution. While, if $n$ is too large, many miss-assigned samples will be contained when estimating the target distribution, because samples are selected based on the clustering label generated by the TSA module. Thus, the performance will be weaken.

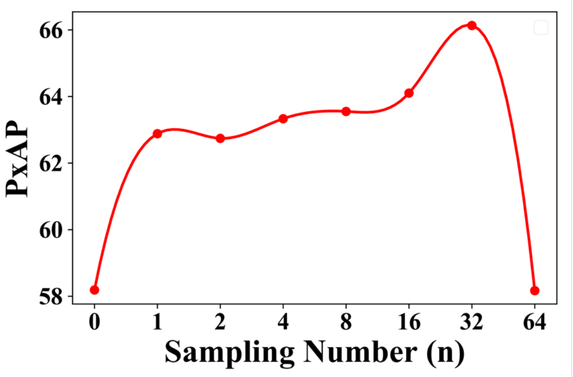

Figure 3. The PxAP using different sample number $n$.

## 2.3. Influence of background threshold

Similar to other WSOL methods, the performance of our DA-WSOL pipeline is also influenced by the background threshold. Thus, except for the threshold-concerned benchmark MaxBoxAcc and PxAP [1], we also plot the IoU under different thresholds, and the precision-recall (PR) curve in Fig. 4. It can be seen that our method has a much higher peak on IoU score, and our PR-curve is located at the upper-right compared with other methods. This shows that our method can better balance the precision and recall when using different background thresholds, which caused our much higher PxAP as indicated in our main paper.

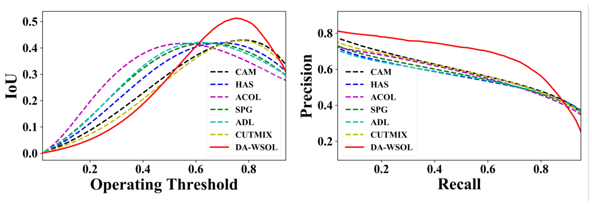

Figure 4. The IoU and PR curve plotted by different threshold.

## 2.4. Results with InceptionV3 backbone

Due to the page limitation, in our main paper, we only give the comparison with SOTA methods with InceptionV3 backbone on OpenImages dataset. Here, we shows our results on ImageNet and CUB-200 dataset with InceptionV3 backbone. Specifically, SGD optimizer with weight decay $5e$-4 and momentum 0.9 is used. For the ImageNet dataset, the initial learning rate $0.001$ is adopted to train our method total 10 epochs, which is then divided 10 times every 3 epochs. Hyper-parameter $\lambda_1$ and $\lambda_2$ are set 0.1 and 0.3, respectively. For the CUB-200 dataset, the initial learning rate are set $0.02$, which is divided 10 times every 15 epochs. The training process is ended at 50 epoch and the two hyper-parameters are set as $\lambda_1 = 1.4$ and $\lambda_2 = 4$. Corresponding results are shown in Table. 2. It can be seen that the results are in accord with the ResNet50 in our paper that our method outperforms SOTA methods in localization score for the ImageNet dataset, while the Top-1 score is a bit lower due to the side-affect of DA. Moreover, for the CUB-200 dataset, our method outperforms the methods that generate class-awareness localization maps (method without underline), which is the same as ours.

Table 2. Comparison between our method and SOTA methods on ImageNet and CUB-200 datasets with InceptionV3 backbone.

| | ImageNet dataset | | CUB-200 dataset | |
|---|---|---|---|---|
| Method | Top-1 Loc | GT-known | Top-1 Loc | GT-known |
| CAM [14] | 46.29 | - | 43.67 | - |
| ADL [2] | 48.71 | - | 53.05 | - |
| DGL [9] | 52.23 | 68.08 | 50.50 | 67.64 |
| I2C [12] | 53.11 | 68.50 | 55.99 | 72.60 |
| ICLCA [5] | 49.30 | 65.21 | 56.10 | 67.93 |
| UPSP [8] | 52.73 | 68.33 | 53.58 | - |
| PSOL [11] | **54.82** | 65.21 | 65.51 | - |
| SEM [13] | 53.04 | 69.04 | - | - |
| FAM [7] | 55.24 | 68.82 | **70.76** | **87.25** |
| GCNet [6] | 49.06 | - | - | - |
| Ours | 52.70 | **69.11** | 65.95 | 80.03 |

∗ *Scores in **bold style** indicate the best.*

∗ *Methods with underline generate class-agnostic map.*

## 2.5. Balance between localization and classification

As discussed in our limitation, adopting DA takes the side-effect on the accuracy of the source domain, which weakens the classification-related metric. Thus, here we explore the balance between localization (target domain) and classification (source domain) by setting different strengths of $\lambda_1$ for our DAL loss. Experiments are conducted on the fine-grained CUB-200 dataset, which is more challenged in the classification task on the source domain. Corresponding results are shown in Fig. 5. It shows that when focusing more on minimizing the distribution between source and target domain (higher $\lambda_1$), the classification accuracy (source accuracy) will keep dropping and the localization accuracy (target domain) firstly increases and then drops. This is because much higher $\lambda_1$ will reduce the influence of the classification loss $\mathcal{L}_c$, which weakens the strength of the source-learned estimator. Thus, both accuracy on source and target domain are weakened.

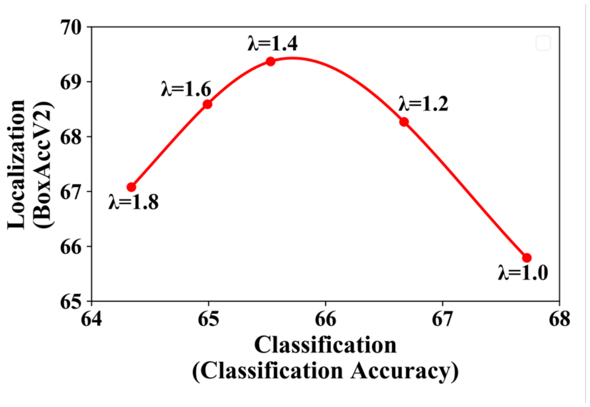

Figure 5. The classification and localization performance of our DA-WSOL plotted by using different $\lambda_1$.

## References

[1] Junsuk Choe, Seong Joon Oh, Seungho Lee, Sanghyuk Chun, Zeynep Akata, and Hyunjung Shim. Evaluating weakly supervised object localization methods right. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3133–3142, 2020. 3

[2] Junsuk Choe and Hyunjung Shim. Attention-based dropout layer for weakly supervised object localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2219–2228, 2019. 3

[3] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015. 2

[4] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012. 2

[5] Minsong Ki, Youngjung Uh, Wonyoung Lee, and Hyeran Byun. In-sample contrastive learning and consistent attention for weakly supervised object localization. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 3

[6] Weizeng Lu, Xi Jia, Weicheng Xie, Linlin Shen, Yicong Zhou, and Jinming Duan. Geometry constrained weakly supervised object localization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, pages 481–496. Springer, 2020. 3

[7] Meng Meng, Tianzhu Zhang, Qi Tian, Yongdong Zhang, and Feng Wu. Foreground activation maps for weakly supervised object localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3385–3395, 2021. 3

[8] Xingjia Pan, Yingguo Gao, Zhiwen Lin, Fan Tang, Weiming Dong, Haolei Yuan, Feiyue Huang, and Changsheng Xu. Unveiling the potential of structure preserving for weakly supervised object localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11642–11651, 2021. 3

[9] Chuangchuang Tan, Guanghua Gu, Tao Ruan, Shikui Wei, and Yao Zhao. Dual-gradients localization framework for weakly supervised object localization. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1976–1984, 2020. 3

[10] Jason Weston, Ronan Collobert, Fabian Sinz, Léon Bottou, and Vladimir Vapnik. Inference with the universum. In *Proceedings of the 23rd international conference on Machine learning*, pages 1009–1016, 2006. 1

[11] Chen-Lin Zhang, Yun-Hao Cao, and Jianxin Wu. Rethinking the route towards weakly supervised object localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13460–13469, 2020. 3

[12] Xiaolin Zhang, Yunchao Wei, and Yi Yang. Inter-image communication for weakly supervised localization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX 16*, pages 271–287. Springer, 2020. 3

[13] Xiaolin Zhang, Yunchao Wei, Yi Yang, and Fei Wu. Rethinking localization map: Towards accurate object perception with self-enhancement maps. *arXiv preprint arXiv:2006.05220*, 2020. 3

[14] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 3