Scaling Novel Object Detection with Weakly Supervised Detection Transformers

Tyler LaBonte[1,2] Yale Song[1] Xin Wang[1] Vibhav Vineet[1] Neel Joshi[1]
[1]Microsoft Research [2]Georgia Institute of Technology
tlabonte@gatech.edu, {yalesong, wanxin, vivineet, neel}@microsoft.com

Abstract

Weakly supervised object detection (WSOD) enables object detectors to be trained using image-level class labels. However, the practical application of current WSOD models is limited, as they operate at small scales and require extensive training and refinement. We propose the Weakly Supervised Detection Transformer, which enables efficient knowledge transfer from a large-scale pretraining dataset to WSOD finetuning on hundreds of novel objects. We leverage pretrained knowledge to improve the multiple instance learning framework used in WSOD, and experiments show our approach outperforms the state-of-the-art on datasets with twice the novel classes than previously shown.

1. Introduction

Object detection is a fundamental task in computer vision where supervised neural networks have demonstrated remarkable performance [4, 19, 24, 26]. A major factor in the success of these approaches is the availability of datasets with fine-grained bounding box annotations [9,11,15,16,18, 28]. However, in comparison to image classification, the annotation process for object detection is considerably more expensive and time-consuming [23]. We consider weakly supervised object detection (WSOD), which aims to learn object detectors using only image-level class labels.

Previous WSOD models [2, 29] generate object proposals using a low-precision high-recall heuristic [31], then use multiple instance learning (MIL) [7, 21] to recover high-likelihood proposals. Another effective strategy leverages a *source dataset* with bounding box annotations to transfer semantic (class-aware) [3, 30] or class-agnostic [32, 35] knowledge to a *target dataset* of novel objects.

Though the presence of many classes in the source dataset is posited to be essential for effective transfer [32], current WSOD methods focus on datasets with few classes such as VOC and COCO-60 [9, 18]. This has two major drawbacks which limit the usage of WSOD models in applications. First, knowledge transfer is most effective when objects in the target dataset have visually similar counterparts in the source dataset [35], so training on few classes may limit the domains where transfer is effective. Second, current WSOD models perform best with multiple rounds of training and refinement [29, 35], requiring additional computation and human identification of optimal parameters.

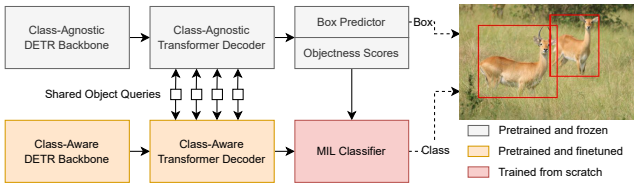

Figure 1. WS-DETR is a two-stream approach utilizing a class-agnostic DETR as proposal generator and a class-aware DETR for weakly supervised finetuning. The two streams share object queries. The MIL classifier leverages objectness knowledge from the pretrained class-agnostic DETR to detect novel objects.

To improve scalability of WSOD, we leverage the pretraining-finetuning approach which has become standard in computer vision [5, 12] – specifically, by pretraining a single detection model on a large-scale, annotated source dataset with hundreds of classes, and then using this model for weakly supervised finetuning on novel objects. Transformer-based methods are particularly well-suited to this problem: while they lack the inductive bias of CNNs, they excel at large-scale training and transfer learning for vision tasks [4, 8]. We propose the Weakly Supervised Detection Transformer (WS-DETR), which integrates DETR [4] with an MIL architecture for scalable WSOD finetuning on novel objects (detailed in Figure 1).

2. Weakly Supervised Detection Transformer

Existing MIL architectures [2] are primarily based on a two-stage RCNN-like [26] structure with region-of-interest (ROI) pooling performed on proposals and the resultant features used for classification; WS-DETR combines the comprehensive proposals of this two-stage framework with the

scalability and simplicity of Transformers. Instead of ROI pooling, we set the WS-DETR object queries to the frozen class-agnostic queries. Thus, the Transformer decoder attends to the same locations as the class-agnostic model and the MIL predictions correspond to object proposals.

2.1. DETR Pretraining

We perform fully supervised pretraining on the source dataset with two models: a class-agnostic DETR trained on binary object labels serves as the proposal generator, while a class-aware DETR trained with full labels provides weight initialization for WSOD. These models extend Deformable DETR [36] and return N object proposals. In the class-agnostic model, a two-layer ReLU network returns proposal coordinates $\{\boldsymbol{p}_i\}_{i=1}^N$ and a fully-connected layer returns classification logits $\{s_i\}_{i=1}^N$ interpreted as objectness scores. In contrast, the class-aware model has two C-class fully-connected layers for classification and detection.

2.2. MIL Classifier

The MIL classifier receives the classification logits $\mathbf{C} \in \mathbb{R}^{N\times C}$ and detection logits $\mathbf{D} \in \mathbb{R}^{N\times C}$ and converts them to an image-level classification prediction. The classification logits are softmaxed over classes, while the detection logits are softmaxed over detections [2]. Let σ denote the softmax operation for $\boldsymbol{z} \in \mathbb{R}^N$: $\sigma(\boldsymbol{z})_i = \frac{\exp z_i}{\sum_{j=1}^N \exp(z_j)}$. We define the class-wise and detection-wise softmaxes as $\sigma^c_{ij}(\mathbf{A}) = \sigma((\mathbf{A}^\top)_j)_i$ and $\sigma^d_{ij}(\mathbf{A}) = \sigma(\mathbf{A}_i)_j$ where $\mathbf{A}_i$ is the i^{th} row of $\mathbf{A}$. The softmaxed matrices for MIL are $\sigma^c(\mathbf{C})$ and $\sigma^d(\mathbf{D})$. These matrices are then multiplied elementwise and summed over detections to obtain the image-level MIL predictions $\hat{y}_j = \sum_{i=1}^N \sigma^c_{ij}(\mathbf{C})\sigma^d_{ij}(\mathbf{D})$. Finally, the negative log-likelihood loss is computed between the MIL predictions and the image-level class labels $\{y_j\}_{j=1}^C$ as $\mathcal{L}_{\text{MIL}} = -\frac{1}{C}\sum_{j=1}^C y_j \log(\hat{y}_j) + (1-y_j)\log(1-\hat{y}_j)$.

The state-of-the-art method for knowledge transfer from a pretrained class-agnostic model is the objectness regularization technique of [35], which uses the class-agnostic objectness scores to regularize the detection branch. Let $S(x) = 1/(1+e^{-x})$ denote the sigmoid operation for $x \in \mathbb{R}$, then $\mathcal{L}_{\text{obj}} = \frac{1}{N}\sum_{i=1}^N \left(\max_{1\le j\le C} S(\mathbf{D}_{ij}) - S(s_i)\right)^2$. The model loss is $\mathcal{L} = \mathcal{L}_{\text{MIL}} + \lambda\mathcal{L}_{\text{obj}}$ for $\lambda > 0$. During inference, WS-DETR returns box $\boldsymbol{p}_i$ with class prediction and confidence determined by $\operatorname{argmax}_{1\le j\le C} \sigma^c_{ij}(\mathbf{C})\sigma^d_{ij}(\mathbf{D})$.

2.3. Joint Probability Estimation

We show in Section 3.2 that objectness regularization [35] is insufficient for general WSOD, as it can suffer from the common MIL weakness of overfitting to distinctive classification features of novel objects [29]. To rectify this and better utilize the pretrained DETR, we propose a formulation for the MIL classifier based on the joint object and class probabilities for each proposal [25]. For a given proposal i, let $c_i = \max_{1\le j\le C} \sigma^c_{ij}(\mathbf{C})$ and $d_i = \max_{1\le j\le C} S(\mathbf{D}_{ij})$ be its maximum classification and detection probabilities.

For a given proposal i, the regularizer $\mathcal{L}_{\text{obj}}$ only cares about the value of d_i and not whether its position in the row actually lines up with c_i – that is, whether $\operatorname{argmax}_{1\le j\le C} \sigma^c_{ij}(\mathbf{C}) = \operatorname{argmax}_{1\le j\le C} S(\mathbf{D}_{ij})$. If these values are mismatched, this failure case would result in low confidences for every proposal and essentially sort them by c_i, causing overfitting. Indeed, we observe the WS-DETR confidences using this technique are typically below 0.01.

Instead, the probability of these distinctive features should be diminished by a low objectness probability as the model recognizes that the feature does not represent an entire object. Hence, we compute the joint probability $\mathbb{P}[i^{th}$ proposal is an object and class $j] = \sigma^c_{ij}(\mathbf{C})S(s_i)$. Using normalized probabilities via softmax [2] we obtain the image-level prediction $\hat{y}_j = \sum_{i=1}^N \sigma^c_{ij}(\mathbf{C})\sigma(\boldsymbol{s})_i$. This joint probability technique is mutually exclusive with objectness regularization [35] and does not utilize $\mathcal{L}_{\text{obj}}$.

2.4. Sparsity in the MIL Classifier

The objectness knowledge present in the pretrained DETR can also be leveraged to reduce noise during multiple instance learning – while there are a fixed number of N proposals, the model typically only detects a few with high objectness scores. To focus more on these confident proposals, we propose utilizing sparsity along the detection dimension of the MIL classifier. While there are many sparsity techniques, we choose sparsemax [22] because of its theoretical justification and successful application in previous MIL architectures [34] (though not in WSOD).

Applying sparsemax zeros out low-confidence boxes, increasing emphasis on correct classification of likely proposals. Specifically, sparsemax returns the Euclidean projection of a vector $\boldsymbol{z} \in \mathbb{R}^N$ onto the $(N-1)$-dimensional probability simplex $\Delta^{N-1} = \{\boldsymbol{p} \in \mathbb{R}^N : \mathbf{1}^\top\boldsymbol{p} = 1, \boldsymbol{p} \ge \mathbf{0}\}$ as $\text{sparsemax}(\boldsymbol{z}) = \operatorname{argmin}_{\boldsymbol{p}\in\Delta^{N-1}} \|\boldsymbol{p}-\boldsymbol{z}\|_2^2$. Thus, instead of $\sigma^d_{ij}(\mathbf{D})$ as in Section 2.2, we substitute $\text{sparsemax}(\mathbf{D}_i)_j$. Note that we still have $\hat{y}_i \in (0,1)$ for all i.

3. Experiments

3.1. FSOD Dataset

Our primary dataset for evaluation is the Few-Shot Object Detection (FSOD) dataset [10], designed to test the performance of few-shot learning models on novel objects in a high-diversity setting. The FSOD dataset comprises 1000 classes, with 800 for training and 200 for testing. This split is generated such that the test classes have the largest distance from training classes in a semantic tree, providing a challenging setting for generalization to novel objects.

In contrast to few-shot learning, WSOD requires a target

Table 1. Class-agnostic performance of Faster R-CNN (used by [35]) and DETR methods trained on FSOD-800 and evaluated on each FSOD-200 test split, ignoring classes. We use the codebase of [35], which does not report precision for this task.

Method	mAP	AP50	mAR
Zhong *et al.* [35]	–	–	50.5 ± 2.1
Class-Aware DETR	18.4 ± 1.0	26.9 ± 0.94	62.3 ± 3.0
Class-Agnostic DETR	$\mathbf{30.6 \pm 1.6}$	$\mathbf{43.0 \pm 1.6}$	$\mathbf{65.5 \pm 3.2}$

Table 2. WSOD performance on FSOD-200 splits with FSOD-800 pretraining. Our WS-DETR is initialized with class-agnostic proposal generator and class-aware weights. The supervised DETR is finetuned from the class-aware FSOD-800 checkpoint.

Method	mAP	AP50	mAR
Zhong *et al.* [35]	20.6 ± 0.76	32.7 ± 2.0	34.4 ± 0.43
WS-DETR Base	13.9 ± 1.6	20.0 ± 1.9	60.1 ± 2.4
WS-DETR Sparse	28.5 ± 0.86	$\mathbf{38.5 \pm 0.63}$	$\mathbf{68.0 \pm 4.3}$
WS-DETR Joint	28.6 ± 0.43	37.8 ± 0.87	65.3 ± 1.5
WS-DETR Full	$\mathbf{28.6 \pm 0.25}$	38.2 ± 1.1	67.4 ± 3.9
Supervised DETR	47.7 ± 1.3	64.0 ± 1.0	76.3 ± 1.2

dataset of novel objects. Thus, we utilize FSOD-800 as a source dataset for pretraining and create three random 80/20 train/test splits of FSOD-200 for WSOD. For each metric, we report the mean and 95% CI based on a t-distribution with two *dof*. FSOD-800 has 52,350 images with 147,489 boxes, while the FSOD-200 splits have 11,322 training images with 28,008 to 28,399 boxes and 2,830 testing images with 6,703 to 7,094 boxes. This setting has $4\times$ the source classes and $2\times$ the novel target classes than previous datasets for WSOD with knowledge transfer [27, 32, 35].

In Table 1, we compare the performance of the class-agnostic and class-aware DETR vs. the class-agnostic Faster R-CNN of [35] trained on FSOD-800 and evaluated on each FSOD-200 test split. For the class-aware DETR, we evaluate the boxes only. Both DETR variants outperform the Faster R-CNN and the class-agnostic DETR has much better precision. We show in Section 3.4 that this precision improvement translates to superior WSOD performance, justifying pretraining a class-agnostic model.

We introduce short names for each permutation of WS-DETR with our techniques from Sections 2.3 and 2.4. "Base" refers to our model with the objectness regularization of [35]; "Sparse" refers to "Base" with added sparsity; "Joint" refers to "Base" with joint probability estimation only; and "Full" refers to "Joint" with added sparsity.

In Table 2, we detail the performance of our model on each FSOD-200 split against the state-of-the-art WSOD knowledge transfer baseline of [35] and a supervised DETR upper bound. The addition of either our joint probability technique or sparsity boosts mAP by 14.7 points over WS-DETR Base, achieving a new state-of-the-art performance by 8 mAP. While the method of [35] loses 15 mAR during weakly supervised training, our WS-DETR approach gains 2.5 mAR relative to the class-agnostic pretrained model.

(a) WS-DETR Base (b) WS-DETR Joint

Figure 2. Visualization of how our WS-DETR joint probability technique prevents overfitting to distinctive classification features on the FGVC-Aircraft dataset. The plotted bounding box is the highest confidence detection in the image.

Table 3. WSOD performance on the FGVC-Aircraft dataset with FSOD-800 pretraining. The WS-DETRs using our joint probability technique achieve near-supervised level performance, while the objectness regularization methods underperform due to overfitting to distinctive classification features.

Method	mAP	AP50	mAR
Zhong *et al.* [35]	14.8	28.7	30.5
WS-DETR Base	5.2	8.5	63.4
WS-DETR Sparse	50.6	57.4	93.2
WS-DETR Joint	77.7	83.6	93.4
WS-DETR Full	**79.1**	**85.0**	**94.2**
Supervised DETR	87.1	88.7	97.9

3.2. FGVC-Aircraft Dataset

The FGVC-Aircraft dataset [20] comprises 10,000 images of 100 types of aircraft whose visual characteristics may differ only slightly between classes. It poses a simpler detection problem, as the target objects are large and centered. And, since "airplane" is a FSOD-800 source class, we expect WSOD models to perform well on this task. We show our joint probability formulation achieves this outcome, while the objectness regularization technique utilized in previous work [35] limits detection performance. In particular, previous models overfit to distinctive classification features which are highly localized in fine-grained datasets, a weakness observed by [29] and whose remedy has been the subject of several WSOD studies [14, 29, 33]. These solutions typically involve multiple rounds of box refinement via self-training. In contrast, our method leverages the objectness knowledge from the pretrained model to identify the correct proposal without any extra computation.

In Figure 2, we visualize how our joint probability estimation technique properly bounds the entire aircraft, while objectness regularization [35] overfits to distinctive features. In Table 3, we display the mAP, AP50, and mAR of each model on the FGVC-Aircraft test set and demonstrate that our model achieves near-supervised level performance.

Table 4. WSOD performance on the iNaturalist 2017 dataset with FSOD-800 pretraining. Our WS-DETR is initialized with class-agnostic proposal generator and class-aware weights. The supervised DETR upper bound is finetuned from the same class-aware FSOD-800 checkpoint. The method of Zhong *et al.* [35] did not converge for the subclasses task.

Method	13 Superclasses		2,854 Subclasses	
	mAP	AP50	mAP	AP50
Zhong *et al.* [35]	44.1	76.7	–	–
WS-DETR Base	0.2	0.4	1.7	3.7
WS-DETR Sparse	**61.1**	**79.3**	30.4	38.2
WS-DETR Joint	54.8	70.0	22.1	29.8
WS-DETR Full	60.7	78.7	**35.4**	**43.5**
Supervised DETR	79.2	93.6	51.5	58.8

Table 5. WSOD performance on each FSOD-200 split with FSOD-800 pretraining. We utilize our joint probability technique and no sparsity. The class-aware DETR is pretrained on FSOD-800 with its full 800 classes, while the class-agnostic DETR is pretrained with only binary object labels.

Proposal Generator	Weights Init.	mAP	AP50
Aware	Agnostic	18.0 ± 1.1	24.3 ± 1.4
Aware	Aware	22.1 ± 1.7	29.7 ± 2.3
Agnostic	Agnostic	27.0 ± 1.0	35.8 ± 1.7
Agnostic	Aware	$\mathbf{28.6 \pm 0.43}$	$\mathbf{37.8 \pm 0.87}$

3.3. iNaturalist Dataset

An application for WSOD not captured by current settings is datasets with many classes which require domain-specific knowledge to label. One exemplar is the iNaturalist 2017 dataset [13], a fine-grained species dataset of 500K boxes and 5,000 classes, 2,854 of which have detection annotations. Van Horn *et al.* [13] remark that the bounding box labeling was difficult since only an expert can distinguish all the species; WSOD is a very practical alternative.

In Table 4, we detail the performance of WS-DETR against the state-of-the-art model [35] on the 13 superclasses and 2,854 subclasses in the dataset. The addition of sparsity to our joint probability technique improves results by up to 13.3 mAP. WS-DETR outperforms the state-of-the-art on the superclasses by 17 mAP, significantly improving high-precision WSOD. While the method of Zhong *et al.* [35] did not converge on the subclasses, WS-DETR achieves 75% of supervised performance.

3.4. Ablation Study

In our above experiments, we used a class-agnostic pretrained DETR as the proposal generator and a class-aware pretrained DETR for initialization for WSOD. We can instead initialize with the class-agnostic DETR to halve needed pretraining. In Table 5, we show that the WS-DETR trained with class-agnostic proposal generator and weights initialization only loses 1.6 mAP and 2 AP50 compared to the best model; this suggests the class-agnostic model can learn most necessary features during WSOD finetuning.

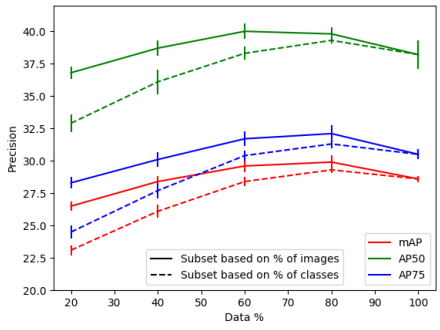

Figure 3. Scaling study of FSOD-800 pretrained WS-DETR Full with FSOD-200 WSOD. We test pretraining with a percentage of images vs. a percentage of classes, then perform WSOD training and evaluate on our held-out test set. This shows pretraining class quantity contributes more to performance than image quantity.

We perform a scaling study on FSOD-800 pretraining with WS-DETR Full and find that class quantity contributes more to downstream WSOD performance than image quantity (see Figure 3). Group 1, the solid lines in the figure, are a random split of FSOD-800 with all classes represented. Group 2, the dashed lines in the figure, have the same number of images as Group 1 but with that same proportion of classes. This experimental setup isolates the effect of increased pretraining classes with the same number of total images. We take 3 random splits of FSOD-800 at each percentage level for each group and finetune on the 3 splits of FSOD-200. We report the mean and 95% CI with respect to a t-distribution with 8 *dof*. This is the first rigorous testing and proof of the hypothesis of Uijlings *et al.* [32] that class quantity is more important than image quantity for WSOD pretraining, and it justifies our usage of FSOD-800 in place of a larger dataset with less classes such as COCO [18]. The lowest proportion of classes we test (160 classes) is still nearly $3\times$ that of COCO-60 [17, 18]; the performance gap at this level suggests that standard datasets used for WSOD pretraining are an order of magnitude too small.

4. Conclusion

We propose Weakly Supervised Detection Transformer (WS-DETR), which integrates DETR with an MIL architecture for WSOD on novel objects. Our model leverages the strengths of both two-stage detectors and the end-to-end DETR framework. In comparison to existing WSOD approaches, which operate at small scales and require multiple rounds of training and refinement, our WS-DETR method utilizes a single pretrained model for knowledge transfer to WSOD finetuning in a variety of practical domains.

References

[1] Amir Bar, Xin Wang, Vadim Kantorov, Colorado J Reed, Roei Herzig, Gal Chechik, Anna Rohrbach, Trevor Darrell, and Amir Globerson. DETReg: Unsupervised pretraining with region priors for object detection. In *35th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 7

[2] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *29th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 2, 7

[3] Tianyue Cao, Lianyu Du, Xiaoyun Zhang, Siheng Chen, Ya Zhang, and Yan-Feng Wang. CaT: Weakly supervised object detection with category transfer. In *18th International Conference on Computer Vision (ICCV)*, 2021. 1

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *16th European Conference on Computer Vision (ECCV)*, 2020. 1

[5] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *7th International Conference on Learning Representations (ICLR)*, 2019. 1

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *22nd IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 7

[7] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1):31–71, 1997. 1

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations (ICLR)*, 2021. 1

[9] Mark Everingham, Luc Van Gool, Christopher K. Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision (IJCV)*, 88(2):303–338, 2010. 1, 7

[10] Qi Fan, Wei Zhuo, and Yu-Wing Tai. Few-shot object detection with attention-rpn and multi-relation detector. In *33rd IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[11] Agrim Gupta, Piotr Dollár, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *32nd IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *29th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 7

[13] Grant Van Horn, Oisin Mac Aodha, Yang Song, Alexander Shepard, Hartwig Adam, Pietro Perona, and Serge J. Belongie. The iNaturalist challenge 2017 dataset. In *31st IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 4

[14] Zeyi Huang, Yang Zou, Vijayakumar Bhagavatula, and Dong Huang. Comprehensive attention self-distillation for weakly-supervised object detection. In *34th Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 3, 7

[15] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision (IJCV)*, 123(1):32–73, 2017. 1

[16] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision (IJCV)*, 128(1):1956–1981, 2020. 1

[17] Seungkwan Lee, Suha Kwak, , and Minsu Cho. Universal bounding box regression and its applications. In *14th Asian Conference on Computer Vision (ACCV)*, 2018. 4, 7

[18] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *13th European Conference on Computer Vision (ECCV)*, 2014. 1, 4, 7

[19] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. In *14th European Conference on Computer Vision (ECCV)*, 2016. 1

[20] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013. 3

[21] Oded Maron and Tomás Lozano-Pérez. A framework for multiple-instance learning. In *10th Conference on Neural Information Processing Systems (NeurIPS)*, 1998. 1

[22] André F. T. Martins and Ramón F. Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *33rd International Conference on Machine Learning (ICML)*, 2016. 2

[23] Dim P. Papadopoulos, Jasper R. R. Uijlings, Frank Keller, and Vittorio Ferrari. Extreme clicking for efficient object annotation. In *16th International Conference on Computer Vision (ICCV)*, 2017. 1

[24] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *29th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[25] Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. In *30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[26] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *18th Conference on Neural Information Processing Systems (NeurIPS)*, 2015. 1

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy,

Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 3

[28] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Jing Li, Xiangyu Zhang, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *17th International Conference on Computer Vision (ICCV)*, 2019. 1

[29] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple instance detection network with online instance classifier refinement. In *30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 3

[30] Yuxing Tang, Josiah Wang, Xiaofang Wang, Boyang Gao, Emmanuel Dellandréa, Robert Gaizauskas, and Liming Chen. Visual and semantic knowledge transfer for large scale semi-supervised object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(12):3045 – 3058, 2018. 1

[31] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and A.W.M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision (IJCV)*, 104(1):154–171, 2013. 1

[32] Jasper R. R. Uijlings, S. Popov, and V. Ferrari. Revisiting knowledge transfer for training object class detectors. In *31st IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 3, 4, 7

[33] Zhaoyang Zeng, Bei Liu, Jianlong Fu, Hongyang Chao, and Lei Zhang. WSOD^2: Learning bottom-up and top-down objectness distillation for weakly-supervised object detection. In *17th International Conference on Computer Vision (ICCV)*, 2019. 3

[34] Lijun Zhang, Srinath Nizampatnam, Ahana Gangopadhyay, and Marcos V. Conde. Multi-attention networks for temporal localization of video-level labels. In *3rd Workshop on YouTube-8M Large Scale Video Understanding, 17th International Conference on Computer Vision (ICCV)*, 2019. 2

[35] Yuanyi Zhong, Jianfeng Wang, Jian Peng, and Lei Zhang. Boosting weakly supervised object detection with progressive knowledge transfer. In *16th European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 3, 4, 7, 8

[36] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *9th International Conference on Learning Representations (ICLR)*, 2021. 2, 7

Table 6. FSOD-800 pretrained WS-DETR performance when freezing or unfreezing the classification layer of the class-agnostic DETR during WSOD finetuning. Freezing this layer gives better objectness transfer, implying that pretraining enables the class-agnostic DETR to learn a transferable representation of objectness.

(a) FSOD-200

Method	Obj. Frozen	mAP	AP50
WS-DETR Joint	✗	28.6 ± 0.43	37.8 ± 0.87
WS-DETR Joint	✓	26.3 ± 0.57	35.0 ± 0.5
WS-DETR Full	✗	$\mathbf{28.6 \pm 0.25}$	$\mathbf{38.2 \pm 1.1}$
WS-DETR Full	✓	26.2 ± 0.7	34.4 ± 0.9

(b) FGVC-Aircraft

Method	Obj. Frozen	mAP	AP50
WS-DETR Joint	✗	73.7	83.6
WS-DETR Joint	✓	**79.1**	84.2
WS-DETR Full	✗	**79.1**	**85.0**
WS-DETR Full	✓	78.7	83.3

(c) iNaturalist Superclasses

Method	Obj. Frozen	mAP	AP50
WS-DETR Joint	✗	54.8	70.0
WS-DETR Joint	✓	54.1	70.9
WS-DETR Full	✗	**60.7**	**78.7**
WS-DETR Full	✓	42.5	54.8

A. Finetuning Objectness Layer

A reasonable question is whether finetuning the classification layer in the class-agnostic DETR leads to increased performance – perhaps the model can use the target dataset to refine its understanding of objectness for a new domain. However, our experiments show this is not the case; as detailed in Table 6, unfreezing the objectness layer decreases mAP in five out of six experiments. This is evidence that that pretraining enables the class-agnostic DETR to learn a transferable representation of objectness.

B. COCO-60 to PASCAL VOC Performance

Though the presence of many classes in the source dataset is posited to be essential for effective transfer [32], previous WSOD methods are typically designed for and trained on datasets with few classes and smaller image sets. A widely-used setting for testing WSOD is PASCAL VOC [9] (20 classes), where knowledge transfer methods use COCO-60 [17, 18] (60 classes) for pretraining, with no class overlap between the 20 and 60 classes.

As we have shown in the main body of our paper, the use of COCO-60/VOC has two major drawbacks which limit the usage of previous WSOD models in practice. Yet, for completeness we have tested our method on this commonly used test case and show the results in Table 7. Our best performing approach is below the leading method by Zhong *et al.* [35]. Recall that in the main body of the paper, we show that our method outperforms Zhong *et al.*'s [35] for diverse datasets with hundreds of novel objects such as FSOD and fine-grained datasets such as iNaturalist and FGVC-Aircraft. We believe this inconsistency in the success of methods between using COCO-60/VOC and FSOD illustrates the benefits of our method, which is the ability to leverage large-scale, diverse pretraining (with FSOD) for weakly supervised detection on large, complex datasets that are common in real-world scenarios. Our results suggest this benefit is due to the scalability of our Detection Transformer model and our novel knowledge transfer method of finetuning an end-to-end detection model instead of a classification model such as ResNet [12, 35]; however, there is a trade-off as Transformer-based methods are known to require much more data. Previous methods work well for the small-scale COCO-60/VOC case but don't handle the large and complex ones well, which we believe are more common in real-world applications. We believe this shows that it is time for WSOD research to move beyond what appears to be an over-optimization to COCO-60/VOC, which is not a useful analogue for real-world datasets, and address the large, complex datasets that we introduce with our work.

Table 7. WSOD performance on PASCAL VOC 2007 with COCO-60 pretraining. The supervised DETR is finetuned from the same COCO-60 checkpoint. The result of Zhong *et al.* [35] includes pseudo ground truth mining.

Method	mAP	AP50	mAR
WSDDN [2]		34.8	
CASD [14]		56.8	
Zhong *et al.* [35]		**59.7**	
WS-DETR Base	18.2	28.4	58.4
WS-DETR Sparse	24.2	36.5	57.7
WS-DETR Joint	23.4	33.8	58.4
WS-DETR Full	23.6	34.2	57.6
Supervised DETR	55.3	77.3	72.7

C. Implementation Details

Our WS-DETR has roughly 40 million parameters. We use $N = 300$ proposals, the default for Deformable DETR. DETR pretraining is conducted using default hyperparameters and the AdamW optimizer. We utilize a Deformable DETR [36] with a ResNet50 backbone [12] initialized from the DETReg [1] ImageNet100 [6] checkpoint. We train DETR on FSOD-800 for 50 epochs at a batch size of 16, dropping the learning rate (*lr*) by a factor of 10 at epoch 40. For WSOD finetuning, the *lr* is 2×10^{-5} for the DETR backbone and input projection, 3×10^{-4} for other DETR parameters including the Transformer, and 1×10^{-3} for the MIL classifier. The weight decay is 1×10^{-4} and we use a dropout rate of 0.1. We perform WSOD training on FSOD-200 and FGVC-Aircraft for 30 epochs, dropping the *lr* after 15 epochs. On iNaturalist, we train with a batch size of 32 for 10 epochs, dropping the *lr* after 8 epochs. DETR pre-

training was performed on 8 V100 GPUs, WSOD training was performed on 4 V100 GPUs, and iNaturalist training was performed on a DGX-2 machine with 16 V100 GPUs; all V100s come in the 32GB configuration.

For the comparisons with Zhong *et al.* [35], we use the publicly available implementation developed by the authors. We use the default hyperparameters of $\beta = 5.0$ and $\lambda = 0.2$ and the SGD optimizer. We train the class-agnostic model on FSOD-800 with a batch size of 4 and *lr* of 4×10^{-3} for 70K steps, dropping the *lr* by 10 after 48K steps. We perform WSOD training on FSOD-200 for 10K steps, dropping the *lr* after 7K steps. For FGVC-Aircraft, we train for 6K steps and drop the *lr* after 4K steps. On iNaturalist, we train with a batch size of 32 and *lr* of 8×10^{-3} for 10K steps, dropping the *lr* after 7K steps.

We observed that the magnitude of objectness regularization in our WS-DETR was much smaller than in [35], and a hyperparameter search established $\lambda = 1000$ as the best default for our architecture. This may suggest that minimizing the regularization term is trivial for a highly overparameterized Transformer architecture. To isolate the performance impact of our WS-DETR method, we do not implement false negative mining.