

Towards Few-Annotation Learning for Object Detection: Are Transformer-based Models More Efficient ?

Quentin Bouniot^{*†}

Angélique Loesch^{*}

Romaric Audigier^{*}

Amaury Habrard^{†‡}

^{*}Université Paris-Saclay, CEA, LIST, F-91120, Palaiseau, France

{firstname.lastname}@cea.fr

[†]Université de Lyon, UJM-Saint-Etienne, CNRS, IOGS,

Laboratoire Hubert Curien UMR 5516, F-42023, Saint-Etienne, France

{firstname.lastname}@univ-st-etienne.fr

[‡]Institut Universitaire de France (IUF)

Abstract

For specialized and dense downstream tasks such as object detection, labeling data requires expertise and can be very expensive, making few-shot and semi-supervised models much more attractive alternatives. While in the few-shot setup we observe that transformer-based object detectors perform better than convolution-based two-stage models for a similar amount of parameters, they are not as effective when used with recent approaches in the semi-supervised setting. In this paper, we propose a semi-supervised method tailored for the current state-of-the-art object detector *Deformable DETR* in the few-annotation learning setup using a student-teacher architecture, which avoids relying on a sensitive post-processing of the pseudo-labels generated by the teacher model. We evaluate our method on the semi-supervised object detection benchmarks *COCO* and *Pascal VOC*, and it outperforms previous methods, especially when annotations are scarce. We believe that our contributions open new possibilities to adapt similar object detection methods in this setup as well.

1. Introduction

Deep learning methods are highly successful when trained on a huge amount of *labeled* data. While gathering data is not difficult in most cases, its labeling is always time-consuming and costly. For instance, labeling medical images requires having access to expert knowledge, while annotating images for dense tasks, like object detection and segmentation in autonomous driving, requires going through a tedious process of drawing polygons or bounding boxes around the objects of interest. A more attractive alternative to this process is considered in our work: to


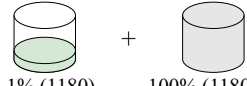
	Few-Shot Learning	Few-Annotation Learning
		
	1% (1180) labeled images	1% (1180) labeled images + 100% (118000) unlabeled images
	Fully Supervised	Semi-supervised (UBT)
FRCNN	9.05%	20.75%
Def. DETR	12.96%	Diverge

Figure 1. Comparison of mean final performance (mAP in %) between Faster R-CNN (FRCNN) [26] and Deformable DETR (Def. DETR) [39] in the Few-Shot and Few-Annotation Learning settings, using only 1% of labeled data on COCO (about 1180 images). See Section 4.1 for experimental details. In the fully supervised case, Def. DETR achieves better results than FRCNN. However, in the semi-supervised case implemented in Unbiased Teacher (UBT) [23], Def. DETR cannot converge.

guide the learning using only a handful of labeled examples, while simultaneously leveraging a large amount of unlabeled data. This corresponds to a particular case of *semi-supervised learning* (SSL) called *few-annotation learning* (FAL) hereafter.

For the task of *Object Detection* (OD), methods in the literature that tackle this setting [15, 30, 23, 35, 37, 31] have all considered object detectors based on traditional convolutional networks [26] with a set of specific post-processing heuristics required for them to work [14, 36]. More recent object detectors are based on an encoder-decoder architecture using transformers [34] that allows for end-to-end OD without depending on this hand-crafted pipeline [5, 39]. However, they have not yet been tested in the SSL context.

The starting point of this paper is the observation that current state-of-the-art transformer-based architecture [39] performs much better than traditional object detectors in a data-scarce fully supervised learning setting, also called *Few-Shot Learning* (FSL), for an equal number of parameters. However, when plugging it into a state-of-the-art *Semi-Supervised Object Detection* (SSOD) method [23], we observe that the model fails to converge, meaning that when used as is (Figure 1), applying SSL methods from literature to transformer-based object detectors does not guarantee good results. Thus, we propose a novel SSL method tailored for transformer-based architectures in order to take advantage of the effectiveness of transformers in FSL, and upscale these methods for FAL. Our proposed method achieves state-of-the-art in several FAL benchmarks.

More precisely, our contributions are summarized as:

- 1) After showcasing the strong performance of transformer-based detectors using few labeled data, we propose *Momentum Teaching DETR* (MT-DETR), an approach for SSOD that leverages the specificities of transformer-based architectures and outperforms previous semi-supervised approaches in FAL settings.
- 2) Contrary to convolution-based OD methods, our approach does not rely on heuristics and post-processing for constructing pseudo-labels. Thus, it eliminates sensitive hyperparameters.

2. Related Work

2.1. Fully Supervised Object Detection

Object Detection is a significant and widely studied problem in computer vision [11, 10, 26, 25, 22, 19, 33]. Essentially, it combines the tasks of object localization and classification. It is a dense task that requires a precise understanding of the image, the objects and their context. The most popular OD models have been based on fully convolutional neural networks [11, 26, 25]. These methods can be separated into *two-stage* [11, 10, 26, 19] or *one-stage* [25, 22, 33] detectors. The former methods make predictions of boxes and their class labels based on region proposals, *e.g.* from a Region Proposal Network (RPN) [26], while the latter make predictions w.r.t. to anchors [20] or a grid of possible object centers [25, 38, 33]. Their performance depends heavily on hand-designed heuristics, with the most prominent example being the Non-Maximal Suppression (NMS) post-processing, widely used in state-of-the-art OD methods [14, 3]. More recently, a novel detector based on an encoder-decoder architecture using transformers [34] has been proposed [5]. This allows end-to-end detection with a simpler pipeline and eliminates the need for the above-mentioned heuristics. The training complexity of this architecture was subsequently improved in Deformable DETR (Def. DETR) [39], by changing the attention operations into deformable attention, which leads to an improved

convergence speed. In this work, we found that Def. DETR is a stronger baseline for FSL than the more popular Faster-RCNN [26] widely used in previous work, which motivated us to focus on transformer-based OD architecture.

2.2. Semi-supervised Learning

The goal of semi-supervised learning is to take advantage of unlabeled data along with labeled data during training. In the more specific case of FAL, it allows reducing the need of a large amount of labeled data by leveraging the use of unlabeled data.

Image Classification The problem of SSL in computer vision was historically tackled first for the image classification task, with significant progress made using deep neural networks [28, 32, 24, 2, 29]. A popular type of approach in this field uses *pseudo-labeling* [17, 2, 1, 29], by generating pseudo-labels from class *predictions* for unlabeled data, either offline [17] or online [2, 29], and then training on a mix of ground truth and pseudo-labels. Another similar branch of methods is using *consistency regularization* [28, 32, 16, 7] to match the predicted class *distributions* of the online version of the model called *student*, to the predicted distributions of a different version of the model called *teacher*, both seeing two *different augmented views* of the inputs. Following recent trends [29, 7], our work takes inspiration from both groups of methods adapted to OD, by training a student model to match the predicted *probability distributions* of proposals made by a teacher model.

Object Detection Methods in the literature are mainly relying on pseudo-labels provided by a teacher model after applying strong data augmentations on unlabeled data [15, 30, 23, 35, 37, 31]. The use of geometric transformations in these strong augmentations is particularly important for OD [30], due to the localization task intrinsic to the problem. The most recent and best performing ones [23, 35, 31] are also updating the teacher through Exponential Moving Average (EMA) [18] of the student’s weights to continuously improve the teacher and, thus, the pseudo-labels given to the student. Although the use of EMA has improved the performance of the models, we propose in our work to stabilize the teacher, by applying an updating strategy throughout training, inspired by recent advances in self-supervised learning [12, 6]. Pseudo-labels are obtained, either by using a *hard labeling* [15, 30, 23, 35, 37] approach, which consists in applying an $\arg \max$ to the predictions, or a *soft labeling* [31] approach, by fully using the predicted distribution. All the previous methods are relying on NMS and thresholding the *confidence scores*, *i.e.* the softmax of the predictions, given by the teacher model. However, the above-mentioned post-processing steps are sensitive to hyperparameters and introduce a bias into the model incentivizing it to be highly confident in its predictions, which may be sub-optimal, particularly when few labeled data are available.

Method	Params.	COCO				VOC07	
		0.5% (590)	1% (1180)	5% (5900)	10% (11800)	5% (250)	10% (500)
FRCNN + FPN [†]	42M	6.83 ± 0.15	9.05 ± 0.16	18.47 ± 0.22	23.86 ± 0.81	18.47 ± 0.39	25.23 ± 0.22
Def. DETR	40M	8.95 ± 0.51	12.96 ± 0.08	23.59 ± 0.21	28.55 ± 0.08	22.87 ± 0.38	29.03 ± 0.46
Δ		+2.12	+3.91	+5.12	+4.69	+4.40	+3.80

Table 1. Performance (mAP in %) comparison between Faster-RCNN (FRCNN) [26] with Feature Pyramid Network (FPN) [19], a two-stage detector commonly used in SSOD methods, and Deformable DETR (Def. DETR) [39], a state-of-the-art transformer-based object detector, with the same ResNet-50 backbone model. The performances are reported for different percentages (and the corresponding number of images) of COCO and VOC07 labeled training data. See Section 4.1 for more details on the experiments. Def. DETR performs better than FRCNN + FPN with fewer labeled data for a similar amount of parameters. [†]: Results from [23] if available, from our reproduction otherwise.

Therefore, we aim to remove all these post-processing steps in this work. Furthermore, SSOD methods in the literature have been exclusively built and evaluated using two-stage OD architectures, and we found that they do not work as is for the more recent detection models based on transformers.

In this paper, we investigate SSOD through the lens of FAL, and we focus our experiments in this setting, in contrast to previous work that address FAL with only a limited number of experiments.

3. A semi-supervised learning approach for transformer-based object detection

In this section, we first motivate our main idea to use a recent state-of-the-art transformer-based OD method in an SSL context by providing several results on both FSL and FAL settings. Then, we present Momentum Teaching DETR (MT-DETR), our transformer-based SSOD method more adapted to FAL and illustrated in Figure 2. More specifically, we describe the construction of the pseudo-labels for unlabeled data, and the update scheduling for the teacher model.

3.1. How do object detectors handle data scarcity ?

From the results presented in Table 1, we can see that Deformable DETR (Def. DETR) [39], a recent state-of-the-art detection model based on transformers, achieves consistently better performance than the most popular two-stage method in FSL. We refer the reader to Section 4.1 for all the implementation details.

These results motivated us to implement Def. DETR in a state-of-the-art SSOD method to see how it performs in FAL settings. We opted for the recent Unbiased Teacher (UBT) [23], as its strong results in FAL were easily reproducible with the provided code. Surprisingly, we observed that with Def. DETR detector, the model does not converge in all the FAL settings tested: 1% of COCO as labeled data (*i.e.* about 1180 labeled images), 5% and 10% of VOC 07 (*i.e.* 250 and 500 labeled images respectively). Even

though it passes by an early best (about 17% mAP on 1% of COCO) at the beginning of training, the model collapses soon after. This diverging behavior is not satisfying in practice, even more so that the same method used with a Faster-RCNN [26] architecture converges (it achieves about 20% final mAP on 1% of COCO) in similar settings (*c.f.* Figure 1). All of this shows that current state-of-the-art SSOD methods are not adapted to more recent transformer-based architectures.

Inspired by these results, we propose an SSL method tailored for transformer-based OD called *Momentum Teaching DETR (MT-DETR)*.

3.2. Overview of our approach

As shown in Figure 2, our approach is composed of a *student-teacher architecture*, which is common for semi-supervised learning [32, 29]. Both student and teacher models are initialized from a fully supervised model trained on the *few labeled data* available. Then, during the semi-supervised training, the method takes as inputs a batch of labeled images $\mathcal{B}^l = \{(x_i^l, y_i^l)\}_{i=1}^{N^l}$ and a batch of unlabeled images $\mathcal{B}^u = \{x_i^u\}_{i=1}^{N^u}$. We define x_i^l and x_i^u as the i^{th} labeled and unlabeled image respectively, $y_i^l = \{y_{(i,j)}^l\}_{j=1}^{k_i} = \{(c_{(i,j)}^l, b_{(i,j)}^l)\}_{j=1}^{k_i} \in \{\{1, 2, \dots, C\} \times \mathbb{R}^4\}_{j=1}^{k_i}$ as the corresponding k_i ground truth class labels and box coordinates, and finally, N^l and N^u are respectively the labeled and unlabeled batch sizes. The student model is updated by a weighted combination of a supervised loss \mathcal{L}_s and an unsupervised loss \mathcal{L}_u with weight $\lambda_u \in \mathbb{R}$:

$$\mathcal{L}(\mathcal{B}^l, \mathcal{B}^u) = \frac{1}{N^l} \mathcal{L}_s(\mathcal{B}^l) + \frac{\lambda_u}{N^u} \mathcal{L}_u(\mathcal{B}^u). \quad (1)$$

Below, we first describe the *supervised branch*, which computes the *supervised loss* using the batch of labeled data \mathcal{B}^l . Then, we detail the *unsupervised branch*, which computes the *unsupervised loss* with the batch of unlabeled data \mathcal{B}^u .

Supervised branch To compute the supervised loss, the supervised branch follows the supervised learning

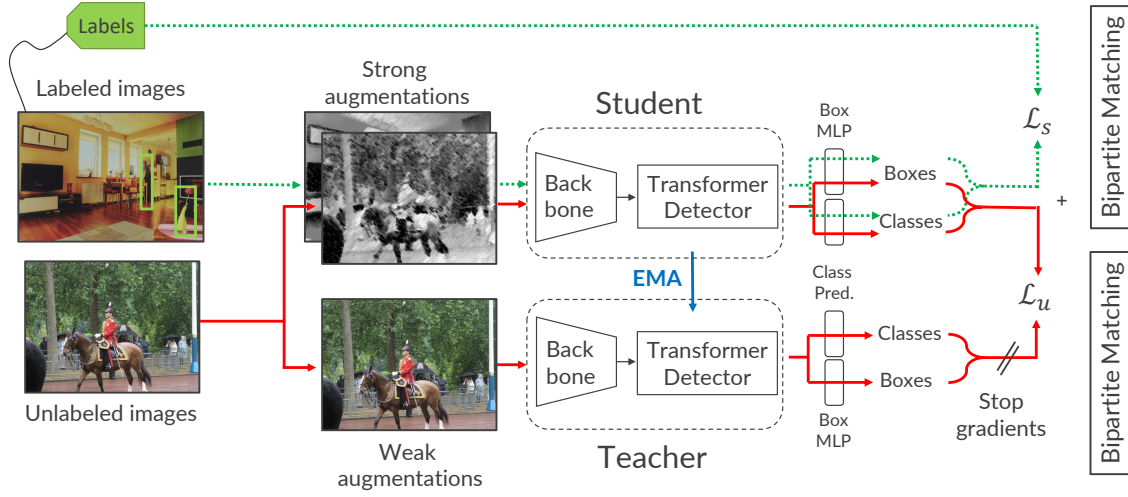


Figure 2. Overview of our Momentum Teaching DETR (MT-DETR) approach for SSOD. The method follows a student-teacher architecture, with the teacher updated through an Exponential Moving Average (EMA) of the student. The keep rate parameter for the EMA follows a *cosine scheduling*. In the supervised branch (in dotted and green), the supervised loss \mathcal{L}_s is computed with the predictions of the student on the labeled images. In the unsupervised branch (in straight and red), the raw, i.e. *unprocessed*, outputs of the teacher model for the weakly augmented unlabeled images are used as *soft* pseudo-labels without applying any heuristic like NMS or confidence thresholding. After finding the best corresponding detection proposals with bipartite matching, the student model learns from the strongly augmented images to match the distribution of class probabilities and the bounding boxes in these pseudo-labels through the unsupervised loss \mathcal{L}_u .

of Def. DETR [39], which is an improved version of DETR [5]. For each image x_i^l , the student model infers N predictions $\hat{y}_i^l = \{\hat{y}_{(i,j)}^l\}_{j=1}^N = \{(\hat{c}_{(i,j)}^l, \hat{b}_{(i,j)}^l)\}_{j=1}^N$ of boxes $\hat{b}_{(i,j)}^l$ and their associated predicted labels *logits* $\hat{c}_{(i,j)}^l \in \mathbb{R}^{C+1}$, with the $(C+1)^{\text{th}}$ logit representing the *no object* (\emptyset) class. Then, the Hungarian algorithm finds from all the permutations of N elements \mathfrak{S}_N , the optimal bipartite matching $\hat{\sigma}_i^l$ between the predictions \hat{y}_i^l of the student model and the ground truth labels y_i^l : $\hat{\sigma}_i^l = \arg \min_{\sigma \in \mathfrak{S}_N} \sum_{j=1}^N \mathcal{L}_{\text{match}}(y_{(i,\sigma(j))}^l, \hat{y}_{(i,\sigma(j))}^l)$. Thus, for each labeled image x_i^l , the j^{th} ground truth $y_{(i,j)}^l$ is associated to $\hat{\sigma}_i^l(j)$. Similarly to the loss used in object detectors, the matching cost $\mathcal{L}_{\text{match}}$ used in the Hungarian algorithm takes into account both class and bounding box predictions through a linear combination of the *Focal loss* [20] $\mathcal{L}_{\text{focal}}$, the ℓ_1 loss of the box coordinates, and the generalized IoU loss [27] $\mathcal{L}_{\text{giou}}$, respectively. These loss functions are then used to compute the supervised loss \mathcal{L}_s as well:

$$\mathcal{L}_{\text{match}}(y_{(i,j)}^l, \hat{y}_{(i,\sigma(j))}^l) = \mathbb{1}_{\{\hat{c}_{(i,\sigma(j))}^l \neq \emptyset\}} \left[\begin{aligned} & \lambda_{\text{class}} \mathcal{L}_{\text{focal}}(c_{(i,j)}^l, \hat{c}_{(i,\sigma(j))}^l) \\ & + \lambda_{\ell_1} \|b_{(i,j)}^l - \hat{b}_{(i,\sigma(j))}^l\|_1 \\ & + \lambda_{\text{giou}} \mathcal{L}_{\text{giou}}(b_{(i,j)}^l, \hat{b}_{(i,\sigma(j))}^l) \end{aligned} \right], \quad (2)$$

$$\mathcal{L}_s(B^l) = \sum_{i=1}^{N^l} \sum_{j=1}^N \left[\lambda_{\text{class}} \mathcal{L}_{\text{focal}}(c_{(i,j)}^l, \hat{c}_{(i,\hat{\sigma}_i^l(j))}^l) + \mathbb{1}_{\{\hat{c}_{(i,\hat{\sigma}_i^l(j))}^l \neq \emptyset\}} \lambda_{\ell_1} \|b_{(i,j)}^l - \hat{b}_{(i,\hat{\sigma}_i^l(j))}^l\|_1 + \mathbb{1}_{\{\hat{c}_{(i,\hat{\sigma}_i^l(j))}^l \neq \emptyset\}} \lambda_{\text{giou}} \mathcal{L}_{\text{giou}}(b_{(i,j)}^l, \hat{b}_{(i,\hat{\sigma}_i^l(j))}^l) \right]. \quad (3)$$

In the above equations, we define $\lambda_{\text{class}}, \lambda_{\ell_1}, \lambda_{\text{giou}} \in \mathbb{R}$ as the coefficients in the matching cost and $\mathbb{1}_{\mathcal{X}}$ the *indicator function*, such that $\forall x, \mathbb{1}_{\mathcal{X}}(x) = 1$ iff $x \in \mathcal{X}$.

Unsupervised branch Our main contribution is the unsupervised loss for transformer-based OD. In the unsupervised branch, we produce two different views for each unlabeled image x_i^u : a *weakly augmented view* $x_i^{u'}$ and a *strongly augmented view* $x_i^{u''}$. Then, the teacher model provides *soft pseudo-labels* $y_i^u = \{y_{(i,j)}^u\}_{j=1}^N = \{(c_{(i,j)}^u, b_{(i,j)}^u)\}_{j=1}^N$, with $c_{(i,j)}^u$ the predicted *logits*, for each *weakly augmented* unlabeled image $x_i^{u'}$, and the student model infers predictions $\hat{y}_i^u = \{\hat{y}_{(i,j)}^u\}_{j=1}^N = \{(\hat{c}_{(i,j)}^u, \hat{b}_{(i,j)}^u)\}_{j=1}^N$ from the corresponding *strongly augmented* unlabeled view $x_i^{u''}$.

We apply the same Hungarian algorithm with the same matching cost $\mathcal{L}_{\text{match}}$ to obtain the best permutation $\hat{\sigma}_i^u = \arg \min_{\sigma \in \mathfrak{S}_N} \sum_{j=1}^N \mathcal{L}_{\text{match}}(y_{(i,j)}^u, \hat{y}_{(i,\sigma(j))}^u)$, that matches the predictions of the student with the closest pseudo-label. In

¹The weak and strong augmentations are described in Section 4.1.

the unsupervised loss \mathcal{L}_u , we follow the consistency regularization paradigm [4, 7, 6]. We train the student network to match the probability distributions of the classes predicted by the student with the soft pseudo-labels proposed by the teacher. We learn to match these distributions by minimizing the cross-entropy between the two class distribution outputs normalized by a softmax function. We define respectively:

$$p_{(i,j)}^s{}^{(k)} = \text{softmax}(c_{(i,j)}^u)^{(k)} = \frac{\exp(c_{(i,j)}^u{}^{(k)})}{\sum_{n=1}^{C+1} \exp(c_{(i,j)}^u{}^{(n)})}, \quad (4)$$

and $p_{(i,j)}^t{}^{(k)} = \text{softmax}(\hat{c}_{(i,j)}^u)^{(k)}$, the student and teacher class distribution outputs, where $c^{(k)}$ is the k^{th} element of c , $\forall c \in \mathbb{R}^{C+1}$. Then the cross-entropy loss is defined as:

$$\mathcal{L}_{\text{CE}}(c_{(i,j)}^u, \hat{c}_{(i,j)}^u) = - \sum_{k=1}^{C+1} p_{(i,j)}^s{}^{(k)} \log p_{(i,j)}^t{}^{(k)}, \quad (5)$$

and finally, we compute the unsupervised loss \mathcal{L}_u as:

$$\begin{aligned} \mathcal{L}_u(\mathcal{B}^u) = & \sum_{i=1}^{N^u} \sum_{j=1}^N \left[\lambda_{\text{class}} \mathcal{L}_{\text{CE}} \left(c_{(i,j)}^u, \hat{c}_{(i,\hat{\sigma}_i^u(j))}^u \right) \right. \\ & + \mathbb{1}_{\{\hat{c}_{(i,\hat{\sigma}_i^u(j))}^u \neq \emptyset\}} \lambda_{\ell_1} \|b_{(i,j)}^u - \hat{b}_{(i,\hat{\sigma}_i^u(j))}^u\|_1 \\ & \left. + \mathbb{1}_{\{\hat{c}_{(i,\hat{\sigma}_i^u(j))}^u \neq \emptyset\}} \lambda_{\text{giou}} \mathcal{L}_{\text{giou}} \left(b_{(i,j)}^u, \hat{b}_{(i,\hat{\sigma}_i^u(j))}^u \right) \right]. \quad (6) \end{aligned}$$

For FAL, we have little information from the labeled data. Therefore, the quality of the pseudo-labels and their contained information play an important part in the training.

3.3. Construction of the pseudo-labels

As mentioned above, the unsupervised loss \mathcal{L}_u takes into account the class predictions through a cross-entropy between the outputs of the student model and the matched outputs of the teacher model. We use the softmax of the outputs of the teacher model as *soft pseudo-labels* for the cross-entropy, as opposed to *hard pseudo-labels* obtained after taking the $\arg \max$.

Following the DETR philosophy [5], we give to the students the *raw soft pseudo-labels* obtained from the teacher, *i.e.* we remove all handmade heuristics to process the teacher outputs, namely, the NMS and confidence thresholding. Both of these post-processing steps are sensitive to hyperparameters and restrict the diversity in the pseudo-labels. By introducing a bias to keep the most confident proposals, they have the unwanted effect of encouraging the models to always be highly confident in their predictions. In the case of FAL, where we have access to only a few labeled examples for each class, the model might not be confident for some classes, leading them to be discarded early

by the post-processing. Relying on the model’s confidence in certain predictions can be tricky. Using the full distributions makes the model less prone to focus on being highly confident in their predictions, and forces the model to take into account the relations between classes. Furthermore, the Hungarian algorithm used in transformer-based OD methods leverages the diversity of proposals given by the model and benefits from the fact that the model is not overconfident on a single class thanks to the matching loss. Indeed, the bipartite matching can favor proposals with better localizations even if the model is less confident in its class predictions, making the use of raw soft pseudo-labels more suitable for transformer-based detectors.

To obtain strong and insightful pseudo-labels helping the student, the teacher must be updated throughout training. We describe the update process in the following section.

3.4. Updating the Teacher model

To avoid a poor supervision from the teacher, its weights θ_t are updated by an Exponential Moving Average (EMA) from the student’s weights θ_s using a keep rate $\alpha \in [0, 1]$:

$$\theta_t \leftarrow \alpha \theta_t + (1 - \alpha) \theta_s. \quad (7)$$

For $\alpha = 1$, the teacher is constant and for $\alpha = 0$ its weights are equal to the student’s. Therefore, there is a trade-off between a too high and too low keep rate parameter. Inspired by the Self-supervised learning literature [12, 6], we update α following a *cosine scheduling* from α_{start} to α_{end} :

$$\alpha \triangleq \alpha_{\text{end}} - (\alpha_{\text{end}} - \alpha_{\text{start}}) \cdot (\cos(\pi k / K) + 1) / 2, \quad (8)$$

with k the current *epoch* and K the maximum number of *epochs*. This scheduling stabilizes the teacher model, especially in the last training iterations, to make it converge at the end of training.

4. Experimental Results

In this section, we present a comparative study of the results of our method to the state-of-the-art on FAL benchmarks, as well as an ablative study on the most relevant parts. Before that, we detail the datasets, the evaluation and training settings used for the different experiments.

4.1. Datasets, evaluation and training details

Datasets and evaluation protocol To evaluate our proposed method, we use the MS-COCO (COCO) [21] and Pascal VOC (VOC) [9] datasets which are standard for object detection, following the settings of existing works [15, 30, 23, 35, 37, 31]. COCO is a dataset with 80 classes, and VOC contains 20 classes. We are specifically interested in two *Few Annotation Learning* (FAL) settings:

On *FAL-COCO*, we randomly sample 0.5, 1, 5 or 10% (respectively about 590, 1180, 5900 and 11800 images) of the

Augmentations	Probability	Parameters	Supervised branch	Unsupervised branch	
				Weak	Strong
Horizontal Flip	0.5	–	✓	✓	✓
Resize	1.0	short edge \in range(480,801,32)	✓	✓	✓
Color Jitter	0.8	(brightness, contrast, saturation, hue) = (0.4, 0.4, 0.4, 0.1)	✓		✓
Grayscale	0.2	–	✓		✓
Gaussian Blur	0.5	$\sigma \in [0.1, 2.0]$	✓		✓
CutOut	0.7	scale $\in [0.05, 0.2]$, ratio $\in [0.3, 3.3]$	✓		✓
	0.5	scale $\in [0.02, 0.2]$, ratio $\in [0.1, 6]$	✓		✓
	0.3	scale $\in [0.02, 0.2]$, ratio $\in [0.05, 8]$	✓		✓
Rotate	0.3	degrees $\in [-30, 30]$			✓
Shear	0.3	shear _x $\in [-30, 30]$, shear _y $\in [-30, 30]$			✓
Rescale + Pad + Translation	0.5	translate _x $\in [0, 0.25]$, translate _y $\in [0, 0.25]$ scale _x $\in [0.25, 0.75]$, scale _y $\in [0.25, 0.75]$			✓

Table 2. The different sets of augmentations used during SSL for each branch. The Horizontal Flip and Resize augmentations follow standard supervised training [5, 39]. The Color Jitter, Grayscale, Gaussian Blur and CutOut augmentations follow Unbiased Teacher [23] training, and the geometric augmentations (Rotate, Shear, Rescale, Pad and Translation) follow Soft Teacher [35] training.

Method	OD Arch.	COCO			
		0.5% (590)	1% (1180)	5% (5900)	10% (11800)
STAC [30]	FRCNN + FPN	9.78 \pm 0.53	13.97 \pm 0.35	24.38 \pm 0.12	28.64 \pm 0.21
Instant-Teaching [37]	FRCNN + FPN	–	18.05 \pm 0.15	26.75 \pm 0.05	30.40 \pm 0.05
Humble Teacher [31]	FRCNN + FPN	–	16.96 \pm 0.38	27.70 \pm 0.15	31.61 \pm 0.28
Unbiased Teacher [23]	FRCNN + FPN	16.94 \pm 0.23	20.75 \pm 0.12	28.27 \pm 0.11	31.50 \pm 0.10
Soft Teacher [35]	FRCNN + FPN	–	20.46 \pm 0.39	30.74 \pm 0.08	34.04 \pm 0.14
MT-DETR (<i>Ours</i>)	Def. DETR	17.84 \pm 0.54 (+8.89)	22.03 \pm 0.17 (+9.07)	31.00 \pm 0.11 (+7.41)	34.52 \pm 0.07 (+5.97)

Table 3. Performance (mAP in %) of our proposed approach on FAL-COCO, using different percentage of labeled data (with the corresponding number of images reported) and 100% of the dataset as unlabeled data. For our method, we also indicate the improvements (in green and in p.p.) w.r.t. the FSL baseline (*c.f.* Table 1).

training set (*train2017*) used as the labeled set and use the full training set for the unlabeled set (about 118k images). Performance is evaluated on *val2017*.

On *FAL-VOC 07-12*, we restrict the labeled training set (VOC07 *trainval*) to a random sample of 5 or 10% (respectively 250 and 500 labeled images), and use the full VOC12 *trainval* (about 11k images) as unlabeled training set. We introduce this novel setting to evaluate our approach in a FAL setting on VOC. We also compare the results with previous SSOD methods using the full VOC07 *trainval* labeled training set (5k labeled images) and VOC12 *trainval* as un-

labeled training set. Performance is evaluated on VOC07 *test* set.

In all settings, performance is reported and compared using the $AP_{50:95}$ (mAP, in %) evaluation metric using the official COCO and VOC evaluation codes, respectively.

Training For a fair comparison, a fully supervised ResNet-50 [13] pretrained on ImageNet [8] is used as a backbone for all the methods. For fine-tuning Def. DETR [39] on the few labeled data, we train the model with a batch size of 32 images on 8 GPUs until the validation performance stops increasing, *i.e.* for COCO, up to 2000 epochs for 1%, 500 epochs for 5%, 400 epochs

Method	OD Arch.	VOC 07-12		
		5% (250)	10% (500)	100% (5000)
STAC [30]	FRCNN + FPN	–	–	44.64
Instant-Teaching [37]	FRCNN + FPN	–	–	50.00
Humble Teacher [31]	FRCNN + FPN	–	–	53.04
Unbiased Teacher [github]	FRCNN + FPN	–	–	54.48
Unbiased Teacher*	FRCNN + FPN	35.98 ± 0.71	40.34 ± 0.95	54.61
MT-DETR (<i>Ours</i>)	Def. DETR	36.95 ± 0.53 (+14.08)	43.15 ± 1.10 (+14.12)	56.2

Table 4. Performance (mAP in %) of our proposed approach on VOC with fully labeled VOC07 and unlabeled VOC12 to compare with previous work, and in the novel FAL-VOC 07-12 settings. Different percentage of VOC07 are used as labeled data (5%, 10% or 100%, with the corresponding number of images reported), and the full VOC12 dataset is used as unlabeled data. For our method, we also indicate the improvements (in green and in p.p.) w.r.t. the FSL baseline (*c.f.* Table 1). * indicates our implementation of Unbiased Teacher [23] in this novel setting to compare with our approach. [github] : updated results after publication [23] taken from their official code released³.

for 10%, and for Pascal VOC, up to 2000 epochs for both 5% and 10%. For semi-supervised learning, we train MT-DETR for 50 (respectively, 250) epochs of the unlabeled data on COCO (respectively, Pascal VOC) with a batch size of 48 labeled images and 48 unlabeled images (respectively, 24 and 24) on 8 GPUs. All experiments with less than 100% of labeled data are reproduced on 3 different random subsets². The training hyperparameters, are defined as in Def. DETR [39]. The coefficients for the losses are set as $\lambda_{\text{class}} = 2$, $\lambda_{\ell_1} = 5$, $\lambda_{\text{giou}} = 2$, and $\lambda_u = 4$. Following the training schedule of Def. DETR, we always decay the learning rates by a factor of 0.1 after about 80% of training. The keep rate parameter α follows a *cosine scheduling* from $\alpha_{\text{start}} = 0.9996$ to $\alpha_{\text{end}} = 1$, with the value of α_{start} chosen according to previous work [23].

When using Unbiased Teacher [23], we follow the official implementation³ and the hyperparameters provided.

Augmentations For strong and weak data augmentations, we follow the common data augmentations used in previous works [30, 23, 35]. We apply a random resizing and random horizontal flip for weak augmentations. We randomly add color jittering, grayscale, Gaussian blur, CutOut patches for strong augmentations and also randomly add rescaling, translation with padding, shearing and rotating as geometric transformations [30] in strong augmentations. In the supervised branch, images are also randomly augmented using weak and strong augmentations without any geometric transformations following Soft Teacher [35] practices. It helps the student model to be augmentation-agnostic, to better predict pseudo-labels coming from non-augmented images in the unsupervised branch. We remove the CutOut augmentation in the supervised branch in the most difficult settings of FAL-COCO 0.5% and 1%, since

it can cover the only labeled small boxes available and is counterproductive. All the parameters for the different augmentations can be found in Table 2.

4.2. Results of FAL on COCO and Pascal VOC

Tables 3 and 4 present the results (mAP in %) obtained by our method compared to previous methods in the literature on the FAL-COCO and FAL-VOC 07-12 benchmarks. As can be seen in both tables, our approach is the only one to consider a transformer-based OD architecture (Def. DETR), as opposed to the commonly used two-stage architecture (FRCNN + FPN). When we implemented Def. DETR into Unbiased Teacher [23] (UBT), we found that the model cannot converge in FAL settings (*c.f.* Figure 1).

First, we can see from both tables that our method always improve performance over the corresponding fully supervised FSL baseline (*c.f.* Table 1). With our method, we outperform state-of-the-art results on all labeled fractions of the dataset, and obtain even more strong results specifically when the annotations are scarce: globally about +1 performance point (p.p.) when using 1k or less labeled images, which is even more significant when the overall performance is low. For FAL-COCO with 1% of labeled images, our method achieves a mean of 22.03 mAP, which is about 1.2 p.p., or 6% of improvement over the state-of-the-art, UBT. Notably, on FAL-VOC with 10% of labeled images, we obtain mean performance of 43.15 mAP, corresponding to 2.81 p.p. or 7% of improvement over UBT. We note that our method also outperform the state-of-the-art when using more labeled data, such as with the 100% labeled VOC07 setting, where we improve of about 1.5 p.p. over UBT.

² See our official repository.

³ Official UBT repository.

Ablative Variant	EMA Scheduling		Initialization		NMS	Confidence Thresholding				mAP (in %)
	Cosine	Constant	After FT	From scratch		∅	0.5	0.7	0.9	
Best	✓		✓			✓				22.25
Abl. Sched.		✓	✓			✓				21.48
Abl. Init.	✓			✓		✓				16.51
Abl. NMS	✓		✓		✓	✓				19.85
Abl. Thresh.	✓		✓				✓			10.26
	✓		✓					✓		17.34
	✓		✓						✓	12.37

Table 5. Ablation studies of the different parts of our method. **Green and bold columns names** indicate a *positive* effect on the performance and **red columns** a *negative* effect. The use of *cosine scheduling*, an *initialization after fine-tuning (FT)* and *raw soft pseudo labels* corresponds to the best combination found.

4.3. Ablation studies

In Table 5, we present an ablation study on the main parts of our approach. We review each ablation below.

EMA scheduling The effect of the EMA scheduling is compared between the *Best* and *Abl. Sched.* rows. We can see that using a *cosine scheduling* to gradually reduce the EMA keep rate parameter α leads to an improvement of about 0.7 p.p., as opposed to using a *constant* value for α as done in other SSL approaches [23, 35, 31].

Initialization In this ablation, we study the effect of *end-to-end semi-supervised learning* [35] in the row *Abl. Init.* which consists in starting the semi-supervised training *from scratch* compared to an initialization *after Fine-Tuning (FT)* in the row *Best*, in which we initialize both student and teacher models from the weights of the fine-tuned model on the few labeled data. As can be seen in Table 5 and contrary to Soft Teacher [35], starting the semi-supervised training from fine-tuned weights is much more effective (about 5.7 p.p. better) than starting from randomly initialized weights, since the teacher model will give useful pseudo-labels to the student from the start of training.

NMS The importance of removing NMS to avoid filtering interesting pseudo-labels and introducing bias is showcased between the rows *Best* and *Abl. NMS*. We can see that, contrary to the common practice when using other detectors [23, 35, 31], the introduction of NMS leads to a performance drop of about 2.5 p.p. This is why we used *raw pseudo labels*, i.e. without any post-processing.

Confidence Thresholding The effect of introducing a threshold to filter out the pseudo-labels given by the teacher with poor confidence is shown in the rows *Best* and *Abl. Thresh.*. We test the results using several common values in the literature (0.5, 0.7 and 0.9) [29, 23, 35]. A value of 0.7 seems to give the best final results (17.34 mAP) between the thresholding variants, but we can see that choosing the

best threshold to apply is extremely sensitive. Similarly to Humble Teacher [31], we also found that removing the confidence threshold to use all the *soft pseudo-labels*, which corresponds to the column with \emptyset , leads to stronger results (22.24 mAP), less sensitivity and fewer hyperparameters.

5. Conclusion

In this work, we experimented in different data scarce settings with the state-of-the-art transformer-based object detector Def. DETR [39] and showed that it performs much better than the most popular two-stage detector Faster-RCNN [26] with FPN [19]. Surprisingly, we found that Unbiased Teacher [23], a state-of-the-art SSOD method, did not converge when applied with Def. DETR.

To address this issue, we propose Momentum Teaching DETR (MT-DETR), an SSL approach tailored for OD based on transformers, in order to leverage their good results with few labeled data. Our method is based on a student-teacher architecture and, contrary to common practice, discards all previously used handcrafted heuristics to process pseudo-labels generated by the teacher. These processing steps are sensitive to hyperparameters, and introduce biases with the unwanted effect of forcing the models to be overconfident in their predictions. We show that our proposed MT-DETR outperforms state-of-the-art methods, especially in FAL settings. Future works could push the data scarcity in OD even further to consider very few labeled examples for each class, and better understand how to match the performance of SSL methods for image classification in this setting [29].

Acknowledgements The authors would like to thank Ievgen Redko for fruitful discussions and proofreading. This work was made possible by the use of the Factory-AI supercomputer, financially supported by the Ile-de-France Regional Council.

References

- [1] Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020.
- [2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32, 2019.
- [3] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-nms—improving object detection with one line of code. In *Proceedings of the IEEE international conference on computer vision*, pages 5561–5569, 2017.
- [4] Cristian Buciluă, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.
- [5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.
- [6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660, 2021.
- [7] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020.
- [8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- [9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.
- [10] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.
- [11] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
- [12] Jean-Bastien Grill, Florian Strub, Florent Althé, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent—a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.
- [13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [14] Jan Hosang, Rodrigo Benenson, and Bernt Schiele. Learning non-maximum suppression. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4507–4515, 2017.
- [15] Jisoo Jeong, Seungeui Lee, Jeessoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. *Advances in neural information processing systems*, 32, 2019.
- [16] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *5th International Conference on Learning Representations, ICLR*, 2017.
- [17] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896, 2013.
- [18] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.
- [19] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.
- [20] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.
- [21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.
- [22] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.
- [23] Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Peizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. In *International Conference on Learning Representations*, 2020.
- [24] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018.
- [25] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.
- [26] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.
- [27] Hamid Rezaatofghi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding

- box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666, 2019.
- [28] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in neural information processing systems*, 29, 2016.
 - [29] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020.
 - [30] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020.
 - [31] Yihe Tang, Weifeng Chen, Yijun Luo, and Yuting Zhang. Humble teachers teach better students for semi-supervised object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3132–3141, 2021.
 - [32] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017.
 - [33] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019.
 - [34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
 - [35] Mengde Xu, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. End-to-end semi-supervised object detection with soft teacher. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3060–3069, 2021.
 - [36] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9759–9768, 2020.
 - [37] Qiang Zhou, Chaohui Yu, Zhibin Wang, Qi Qian, and Hao Li. Instant-teaching: An end-to-end semi-supervised object detection framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4081–4090, 2021.
 - [38] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.
 - [39] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021.