

The Impact of Burn-In on Semi-Supervised Single-Stage Object Detection on Small Military Datasets

Michel van Lier^a, Frank Ruis^a, Thijs Eker^a, and Hugo J. Kuijff^a

^aTNO, Oude Waalsdorperweg 63, 2597 AK, the Hague, the Netherlands

ABSTRACT

Semi-Supervised Object Detection (SSOD) aims to improve object detection performance by leveraging both labeled and (large amounts of) unlabeled data. By reducing the reliance on labeled data, SSOD enables faster model development and quicker deployment to the field, allowing systems to rapidly adapt to new environments or mission-specific scenarios with minimal annotation effort.

While SSOD has demonstrated success in enhancing the performance of object detection models, ranging from two-stage detectors like R-CNN and DETR to more recent one-stage detectors such as YOLO, existing research has primarily focused on commonly studied benchmark datasets like PASCAL VOC and MS COCO. To the best of our knowledge, its effectiveness on domain-specific and smaller military relevant datasets has not been thoroughly evaluated.

In this work, we investigate the performance of SSOD using YOLOv5 on three (military-relevant) use-case datasets: air-to-ground dataset VisDrone2019-DET, ground-to-ground dataset Automatic Target Recognition (ATR) Algorithm Development Image Database, our own in-house recorded and annotated proprietary military air-to-ground dataset, and our own internet crowd sourced Russian-Ukrainian War dataset containing ground-to-ground and air-to-ground imagery. We artificially limit the the labeled training data from 66% up to 1%, while treating the remainder as unlabeled. Our results show that SSOD consistently improves object detection performance across all datasets and label proportions compared to training without unlabeled data. Additionally, we find that the choice of burn-in epochs (the point at which labeled pretraining transitions to semi-supervised training) significantly impacts final performance. The optimal burn-in epoch is not necessarily the best performing validation epoch, highlighting the need for a careful burn-in approach.

Keywords: Semi-Supervised Object Detection, Semi-Supervised learning, YOLO, Ground-to-Ground Imagery, Air-to-Ground Imagery, EDF STORE

1. INTRODUCTION

Artificial intelligence (AI) algorithms can play a key role in supporting security and defense systems, for example by detecting threats in rapidly changing operational environments. Computer vision-based object detection can detect, recognize, and identify objects of interest such as adversary vehicles, drones, weapons, or personnel.^{1,2} Integrating such algorithms into military systems improves situational awareness, accelerates decision-making, and reduces the risk of human error.

State-of-the-art object detection algorithms are usually trained in a supervised manner using large, manually annotated datasets with operationally relevant targets. These models commonly achieve best performance when the training data is balanced amongst classes and closely matches the operational context. However, manual annotation of data is time-consuming, and military deployments often involve new environments, weather conditions, or previously unseen objects. The latter includes evolving camouflage patterns or improvised equipment that are not present in existing training data, as observed in recent conflicts like the Russian-Ukrainian war.³⁻⁹ These new conditions and domain shifts can degrade object detection performance, which poses serious risks when used in security and defense operations. To remain effective, AI-based object detection algorithms must be able to quickly adapt to new domains and (improvised) target types with minimal human intervention. Traditional supervised methods that rely on manual labels are impractical.

Semi-supervised object detection (SSOD) is a promising alternative that combines a small set of labeled data with a much larger pool of unlabeled data.¹⁰ This approach reduces the need for manual labels, while improving

adaptability and performance in new operational settings. Although SSOD has demonstrated strong results on large-scale, generic benchmark datasets such as MS-COCO¹¹ and PASCAL VOC,¹² its potential in smaller, domain-specific datasets such as military-relevant contexts, remains underexplored.

SSOD has been largely driven by teacher–student (mean-teacher¹³) paradigms where a teacher network—often an exponential moving average (EMA) of the student—generates pseudo-labels on unlabeled data, and the student is trained on these labels under strong augmentations.¹⁴ Consistency regularization across weak and strong views, together with pseudo-label filtering or re-weighting, has proven effective in suppressing noise and improving label quality.¹⁴ These strategies have been successfully applied to one-stage detectors (YOLO¹⁵), two-stage detectors (Faster R-CNN¹⁶), and transformer-based architectures such as DETR¹⁷ and DINO.¹⁸ While methods such as Unbiased Teacher,¹⁹ Soft Teacher,²⁰ and Semi-DETR²¹ have reported substantial performance gains. A key limitation remains efficiency as conventional teacher models are computationally expensive. This motivates approaches like Efficient Teacher, which seek to preserve pseudo-label quality at lower computation costs. In parallel, the emergence of foundation models (e.g., OWL-ViT²²) offers powerful alternatives that combine detection with broader generalization capabilities. A systematic comparison of SSOD methods such as Efficient Teacher with these foundation models is therefore crucial to assess their relative strengths, limitations, and future relevance in real-world applications.

In this work, we evaluate the effectiveness of SSOD and the effect of burn-in using YOLOv5 on four smaller, domain-specific, military-relevant datasets:

- VisDrone2019-DET:²³ A drone-captured civilian dataset containing common urban objects from an air-to-ground perspective.
- Automatic Target Recognition Algorithm Development Image Database (ATR):²⁴ A ground-to-ground dataset with day and night imagery of military vehicles captured using visual and Mid-Wave Infrared (MWIR) sensors.
- Proprietary military air-to-ground dataset (TNO-A2G): An in-house annotated dataset containing military vehicles.
- Ukraine-Russia War dataset: An crowd-sourced, in-house annotated dataset containing ground-to-ground and air-to-ground imagery of personnel and military vehicles.

For the first three datasets, we artificially constrain the amount of labeled data, while for the fourth we leverage a large pool of unlabeled crowd-sourced imagery. This enables us to analyze the effect of burn-in epochs (the transition point from supervised to semi-supervised training) on object detection performance. Our results show that SSOD consistently improves accuracy across all datasets, particularly when labeled data is scarce, demonstrating its potential to accelerate deployment in operational environments. However, we also find that the optimal burn-in duration is not necessarily aligned with the best supervised checkpoint, making burn-in scheduling a critical factor in Efficient Teacher. To address this uncertainty, we explore model soups, which provide a robust starting point without extensive hyperparameter tuning.

2. METHODS

SSOD often follows a teacher–student approach in which a teacher object detection model, typically an EMA copy of the student object detection model, generates pseudo-labels for unlabeled data. The student is trained on both labeled data (using ground-truth labels) and unlabeled data (using the teacher’s pseudo-labels). This approach allows the student to benefit from large quantities of unlabeled images, but can be sensitive to pseudo-label noise and label confidence thresholding.

In this work the Efficient Teacher SSOD framework is used,¹⁰ which follows a teacher–student learning paradigm (Figure 1). While teacher–student SSOD approaches have previously been applied to multi-stage object detectors, Efficient Teacher is the first to adapt this strategy specifically for the single-stage YOLOv5 architecture.

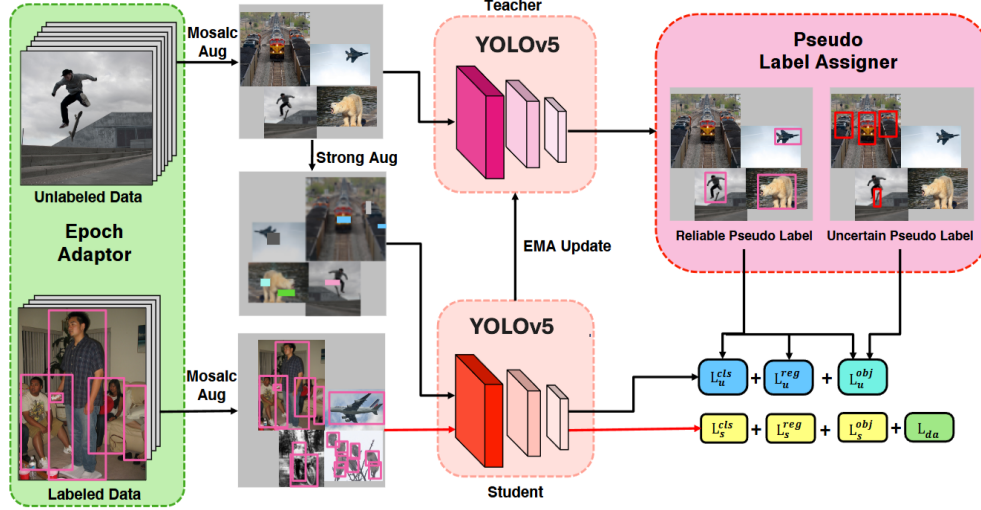


Figure 1: Overview of the Efficient Teacher framework using YOLOv5 (adapted from¹⁰). The framework introduces three key contributions: (1), SSOD with a one-stage YOLOv5 detector, (2) Pseudo-Label Assigner (PLA), which categorizes pseudo-labels into two types to reduce inconsistency, and Epoch-Adaptor (EA), which mitigates the instability of fixed confidence thresholds by dynamically adjusting them per class and per epoch.

YOLOv5 is a widely used single-stage, anchor-based object detector that directly predicts bounding boxes and class probabilities from dense feature maps in a single forward pass without an intermediate proposal generation stage.¹⁵ Compared to two-stage detectors such as Faster R-CNN,¹⁶ which first produce region proposals before refinement, single-stage detectors typically achieve higher inference speed at lower computational cost. This makes them suitable for real-time and resource-constrained applications. However, their dense prediction nature makes them more susceptible to pseudo-label noise in SSOD: inaccuracies in pseudo-label confidence can lead to a high number of false positives.

To address these challenges, Efficient Teacher introduces two additional components:

1. Pseudo-Label Assigner (PLA): a mechanism to improve pseudo-label quality by refining the matching process between predicted labels and ground-truth labels.
2. Epoch Adaptor (EA): a dynamic scheduling mechanism that adjusts training parameters over time to better balance the supervised and semi-supervised learning phases.

The PLA mitigates pseudo-label noise by splitting teacher predictions into reliable and uncertain sets (Figure 2). The high and low thresholds (τ_1 and τ_2 respectively) of the pseudo-label score are used to determine these two types of pseudo-labels. Reliable predictions that have high classification confidence $\geq \tau_1$ are treated as hard supervision labels. Uncertain predictions with $\tau_1 < \text{confidence} \leq \tau_2$ are not discarded, but guide learning via a soft consistency loss which encourages the student's output distribution to align with the teacher's without enforcing a single hard label. This selective supervision reduces the risk of propagating incorrect pseudo-labels while still leveraging useful information from uncertain detections.

The EA addresses the instability of fixed pseudo-label thresholds by dynamically adapting confidence thresholds on a per-class and per-epoch basis. In early training epochs, thresholds are conservative to avoid low-quality pseudo-labels. As the model improves, thresholds are relaxed for classes where the teacher shows stable performance. Equations 1 and 2 show EA threshold calculation for τ_1 , τ_2 , where N_l , N_u are the total labeled and unlabeled samples, N_c^k is the number of ground-truth labels for class c within the labeled samples at epoch k . P_c^k is the sorted list of pseudo-label confidence scores for class c at epoch k (sorted as highest to lowest). The first term in Equations¹⁰ 1 and 2 takes the confidence score at the position corresponding to that expected count

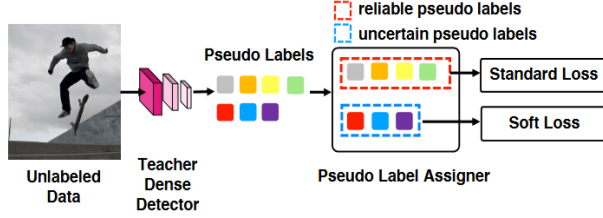


Figure 2: Example of division of pseudo-labels (adapted from¹⁰).

and marks the point confidence score above which pseudo-labels are considered reliable. Additionally, during the burn-in phase, EA applies domain adaptation to align features between labeled and unlabeled data, mitigating domain shift before pseudo-labels are used in the SSOD phase. This is achieved by adding a Gradient Reversal Layer (GRL) and a domain classifier on each of the three output feature maps of YOLOv5. During the burn-in phase, this domain adaptation approach minimizes the domain gap while maximizing the domain classification. This dynamic adaptation ensures balanced supervision across classes and training stages.

$$\tau_1^k = P_c^k \left[n_c^k \cdot \frac{N_u}{N_l} \right] \quad (1)$$

$$\tau_2^k = P_c^k \left[\alpha\% \cdot n_c^k \cdot \frac{N_u}{N_l} \right] \quad (2)$$

In short the Efficient Teacher training is executed in two stages:

- **Burn-in Phase.** In the burn-in phase, the student and teacher are trained in a supervised way on labeled data only, with EA-driven domain adaptation aligning labeled and unlabeled feature distributions.
- **SSOD Phase.** The SSOD phase continues training from the last checkpoint of the burn-in. The teacher produces pseudo-labels for unlabeled data, which is processed by PLA and filtered with EA’s adaptive thresholds. Reliable predictions drive hard detection loss and uncertain predictions contribute via soft consistency regularization.

3. EXPERIMENTS

To evaluate the impact of burn-in on semi-supervised single-stage object detection, we conducted a series of experiments across multiple military-relevant datasets. These experiments were designed to assess the effectiveness of the Efficient Teacher framework under varying conditions, including different ratios of labeled data as well as diverse contexts. We also examined the influence of burn-in duration on subsequent SSOD performance and explored model soups to mitigate variability.

3.1 Experimental Setup

3.1.1 Datasets

VisDrone2019-DET VisDrone2019-DET is a civilian dataset featuring 10,209 daytime images recorded from a drone with a visual sensor overseeing urban and suburban areas in China. The dataset is annotated with 2.6 million bounding boxes across eleven object classes: pedestrian, people, bicycle, car, van, truck, tricycle, awning-tricycle, bus and motor. The dataset poses challenges such as small objects, dense distributions, occlusions and complex backgrounds, making it a challenging benchmark for civilian aerial object detection. We use the official VisDrone2019-DET splits for training, validation and test. Figure 3 provides an impression of this dataset.



Figure 3: An impression of the VisDrone2019-DET dataset, taken from the validation set.²³

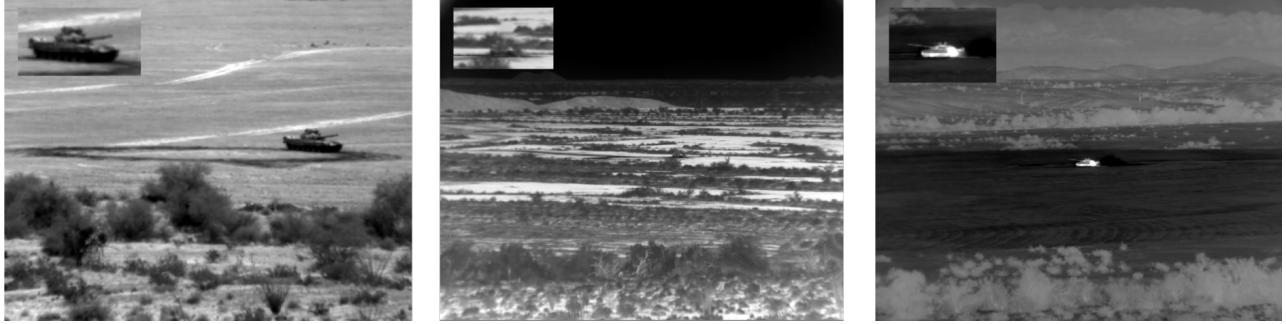


Figure 4: An impression of the ATR dataset taken from our validation set. From left to right: an image at 1000 m distance with visual imagery, an image at 2500 m distance with MWIR imagery during the day and an image at 1000 m distance with MWIR imagery during the night.

ATR The Automatic Target Recognition Algorithm Development Image Database (ATR) is a ground-based object detection dataset containing video recordings of both military and civilian vehicles and persons, captured during day and night.²⁴ The dataset includes data captured with both visual and MWIR sensors. It consists of eleven object classes: pickup, SUV, BTR70 (Bronetransportyor, a Soviet eight-wheeled armored personnel carrier), BRDM2 (Boyevaya Razvedyvatelnaya Dozornaya Mashina, a Soviet amphibious armored scout car), BMP2 (Boyevaya Mashina Pekhoty, a Soviet amphibious infantry fighting vehicle), T62 (Soviet main battle tank), T72 (Soviet main battle tank), ZSU23-4 (Zenitnaya Samokhodnaya Ustanovka, a Soviet self-propelled anti-aircraft weapon system), 2S3 (Soviet self-propelled gun), MTLB (Mnogotselevoy Tyagach Legky Bronirovanny, a multipurpose tracked armored vehicle that was towing a D-20 152mm howitzer), and person. Each vehicle was recorded while driving in a circle with a 100 m diameter at approximately 16 km/h at varying distances from the cameras, from 1000 m to 5000 m in 500 m increments. Recordings were conducted during the day (visible light and MWIR) and night (MWIR only), with each video lasting around 72 seconds at a frame rate of 25 fps. The dataset poses challenges such as small objects, blurred objects, day/night time, and two sensors. Figure 4 provides an impression of this dataset. In this work, a subset of the original dataset was used. The T62 class was excluded because of its infrequent representation, caused by a mechanical failure.²⁴ Additionally, distance ranges exceeding 3000 m were omitted, since objects become too small for reliable detection; persons were also excluded for this reason. Because of the low speed and high frame rate, many consecutive video frames appeared similar. Thus, videos were subsampled by selecting every 50th frame, resulting in 36 frames per distance, object class, time of day (day/night), and sensor type (visible/MWIR). Since the ATR dataset does not provide bounding boxes required for object detection, those defined in our previous publication were used.²⁵ The dataset also lacks an official train/validation split. Therefore, all classes, times of day, and sensors are included in the training set, which is divided based on distance: 1000 m, 1500 m, 2500 m, and 3000 m distances are used for training, while 2000 m is selected as the validation set. This results in an 80%-20% train-validation split, aligning with standard practices in object detection research.

TNO-A2G The TNO-A2G is our proprietary in-house recorded and annotated military air-to-ground dataset. The datasets consists of 31 videos, covering multiple hours, recorded from an two airplane types with a Wescam MX10 and Wescam MX15 visual camera's at 4,000-6,000 feet altitude and a drone with a visual camera at



Figure 5: An AI generated impression of the TNO-A2G dataset. Because of the classification of the dataset, actual images cannot be shown. With respect to the AI generated images, the original images contain more noise, blur, and a wider variation of object sizes.

Table 1: Overview of the TNO-A2G subsets, where videos are grouped in a videoset based on the used aircraft, recording location(s), and moment in the year. Recording locations are Marnehuizen or 't Harde, both in the Netherlands, and a location in Germany. Recordings were made in spring, summer and autumn. Videoset 1 is used as validation set. The other three videosets are used for training.

Videoset	Airplane	Location	Moment in year
1	Airplane	Marnehuizen	Autumn
2	Airplane	't Harde	Summer
3	Drone	Marnehuizen, 't Harde, Germany	Spring and Summer
4	Airplane	't Harde	Summer

50-200 meter altitude, both with adjustable zoom. The bird’s-eye footage covers multiple military exercises in wooded and open areas across several months during spring, summer and autumn, at three locations in two countries (Marnehuizen and 't Harde in the Netherlands, and a location in Germany) during daytime. The labels include the following six object classes: person, PzH 2000 (armored howitzer), Boxer MRV (multi-role armored vehicle), Fennek (armored vehicle), trucks, and civilian vehicles. The dataset presents challenges such as varied bird’s-eye viewpoints, small and camouflaged objects, partial occlusions by vegetation or buildings, image noise and blur. To ensure sufficient visual variation between frames, we selected frames at intervals of 2 to 4 seconds, which resulted in a total of 3988 annotated frames with 11149 labels. Since no official dataset split is available, we defined our own. The videos were grouped into four subsets based on recording location(s), day(s), and used aircraft. This approach minimizes overlap between the labeled and unlabeled training sets and the validation set, as shown in Table 1. For all the experiments, videoset 1 is used as validation set, while the others were used for training. Because of the classification restrictions on the data, the dataset cannot be publicly released. Therefore, an AI-generated impression is provided in Figure 5.

UA-WAR The UA-WAR dataset consists of videos from the frontlines in Ukraine scraped from war footage channels on Reddit and Telegram, an example of which can be seen in Figure 6. This is a challenging dataset due to the large variety in platform, camera quality, pixels on target, environment, video orientation, video editing effects, and general chaos of a live battlefield. Many targets are small, out of focus, camouflaged, obscured, or have been heavily modified with e.g. steel enclosures welded onto tanks or soldiers wearing thermal cloaks. A total of 5458 frames that have been manually verified to contain soldiers or vehicles have been selected from 1314 videos split into 3 sets: train, val, and unlabeled. 792 frames from 185 videos have been labeled as a train set, 158 frames from 141 different videos have been labeled as a test set, and the remaining 4384 frames from 988



Figure 6: An impression of the UA-WAR dataset, taken from the training set.

different videos are kept unlabeled. The labels are person, (civilian) vehicle, truck and armored/military vehicle.

3.1.2 SSOD Model

The Efficient Teacher approach was evaluated using YOLOv5m on the selected datasets. YOLOv5m is of moderate size, which may reduce the risk of overfitting on these smaller datasets, while maintaining a balance among training and inference speed and object detection performance. MS COCO pretrained weights were used.

Training Hyperparameters The Adam optimizer is used with an initial learning rate of 0.001, and a final learning rate of 0.01 for YOLOv5m. Number of epochs for burn-in and SSOD training are dynamic; early stopping is set to epoch 25, to ensure a stabilized burn-in phase before switching to the SSOD phase. Data augmentation follows the original YOLOv5 ‘scratch-high’ setting, but excluding mixup and copy-paste. Labeled and unlabeled data are sampled equally at 1 : 1 ratio. Batch sizes are set to 16 for VisDrone, ATR, and UA-WAR datasets, and 4 for the TNO-A2G dataset. Model input resolution is 960×960 for VisDrone, 640×640 for ATR and UA-WAR, and 1920×1920 for TNO-A2G. τ_1 and τ_2 are determined via EA as discussed in Section 2. Teacher loss weight is 0.5, domain adaptation loss weight is 0.01, EMA smoothing is 0.999, and other hyperparameters follow the Efficient Teacher defaults.

Evaluation We evaluate the models object detection performance using mean average precision (mAP) with 50% intersection over union (mAP@50), since our focus is on detection and classification accuracy rather than precise bounding boxes.

3.1.3 Foundation Model Baseline

We additionally fine-tune OWLv2-large²⁶ as a closed-set detector using DETR-style set prediction. A fixed set of queries predicts boxes and classes; bipartite (Hungarian) matching assigns predictions to ground-truth boxes. Class prompts are the dataset class names passed through the OWLv2 processor; unmatched queries map to a “no-object” class.

Matcher and Losses During assignment we add the OWLv2 objectness logit to the Hungarian matching cost (as a small bias toward confident regions), while not supervising a separate objectness loss; classification uses a no-object weight (eos_coef=0.1). Matching cost otherwise follows the DETR recipe (classification, ℓ_1 , and GIoU terms). The training loss is

$$\mathcal{L} = \lambda_{\text{cls}} \mathcal{L}_{\text{CE}} + \lambda_{\ell_1} \mathcal{L}_{\ell_1} + \lambda_{\text{GIoU}} \mathcal{L}_{\text{GIoU}}, \quad \lambda_{\text{cls}}=2, \lambda_{\ell_1}=5, \lambda_{\text{GIoU}}=2,$$

with a “no-object” weight eos_coef=0.1 in the classification term.

Training Hyperparameters We train with batch size 2 using AdamW (weight decay 10^{-4}), learning rate 5×10^{-6} , early stopping (patience 5), and 20–40 epochs depending on the split. Image resolution is 640×640 for ATR and 960×960 for VisDrone. Inference uses no Non-Maximum Suppression (NMS); we apply a confidence threshold of 0.2. Furthermore, similar to the SSOD model, mAP@50 is used for evaluation.

Table 2: Summary of total annotation and image counts for each artificially constrained label scenario in the VisDrone2019-DET, ATR, and TNO-A2G datasets and their validation sets. Values are shown as: labels (labeled/unlabeled) and images (labeled/unlabeled).

Labeled	VisDrone2019-DET		ATR		TNO-A2G	
	Labels	Images	Labels	Images	Labels	Images
100%	342,957	6,471	2,844	2,556	11,149	3,988
66%	220,757/122,200	4,206/2,265	1,891/953	1,552/4,407	7,288/3861	2,656/1,332
33%	101,666/241,291	1,941/4,530	953/1,891	851/1,705	3,614/7,535	1,328/2,660
16%	50,007/292,950	970/5,501	469/2,375	424/2,132	1,989/9,160	666/3,322
10%	32,646/310,311	647/5,824	237/2,607	211/2,345	1,158/9,991	399/3,589
5%	16,755/326,202	324/6,147	112/2,732	105/2,451	546/10,603	199/3,789
2%	6,444/336,513	129/6,342	55/2,789	51/2,505	213/10,936	80/3,908
1%	3,254/339,703	65/6,406	29/2,815	26/2,530	116/11,033	40/3,948
Val set	38,606	548	720	648	5,735	1,554

3.2 Artificially constrained Labeled Data Experiment

With the artificially constrained labeled data experiments, we measure the impact of reducing the ratio of labeled to unlabeled data on the object detection performance of Efficient Teacher. The VisDrone2019-DET, ATR, and TNO-A2G datasets were selected, since all their data is fully labeled. The baseline in these experiments is training with 100% of the labeled data. To artificially limit the labeled data and to evaluate Efficient Teacher’s performance under constrained labeled data, we progressively reduce the proportion of labeled training data to 66%, 33%, 16%, 10%, 5%, 2%, and 1%. The remaining portion of the training set is treated as unlabeled data. We assess how effectively Efficient Teacher leverages these unlabeled samples by comparing the best-performing validation set model from the burn-in phase against the best-performing model from the SSOD phase. The percentage proportion of labeled training data is defined based on the number of images. Due to confidentiality of TNO-A2G, the mAP@50 scores cannot be shared, so only the improvement between burn-in and SSOD is provided. Section 4.1 presents and discusses the results. Table 2 summarizes annotation and image counts for each artificially constrained label scenario.

3.3 UA-WAR Experiment

The UA-WAR dataset was collected as a relevant and challenging real-life application for semi-supervised object detection. The data is challenging and unique enough such that there are no publicly available datasets or pre-trained (foundation) models that can reach a satisfying detection performance. It is additionally challenging and time-consuming to accurately label any objects of interest, where even after multiple passes by multiple people over our relatively small training and test sets we would still occasionally find another object that was missed during labeling. This experiment is approached as a challenge of what is the best performance we can reach on this dataset using just a small amount of labeled data.

Since COCO-pretrained weights are extremely out-of-domain for challenging footage from a predominantly drone viewpoint, training on exclusively UA-WAR images quickly leads to over-fitting. To prevent this, we include the VisDrone training set with its labels mapped to that of UA-WAR (for person, vehicle and truck, since VisDrone does not contain military vehicles). The TNO-A2G validation set is used to determine the best model during training, such that the UA-WAR test set can be used purely for calculating test performance.

3.4 Burn-In Stopping Point and Model Soups Experiments

These experiments attempt to discern at which point in the burn-in period it would be optimal to switch to semi-supervised training. A straightforward choice would be switch once performance on the validation set has saturated, but we find that this often actually leads to the least gain or even worse than burn-in performance, compared to making the switch earlier. To find the optimal stopping point, we perform an extended burn-in for

Table 3: Results of the artificially constrained labeled data experiments, showing the burn-in $\rightarrow$ SSOD performance in mAP@50 (and percentage of the burn-in it outperforms). Because of confidentiality, the mAP@50 scores for the TNO-A2G cannot be shared.

Labeled	VisDrone2019-DET	ATR	TNO-A2G
100%	49.6	92.4	
66%	46.7 $\rightarrow$ 46.9 (+0.2)	89.9 $\rightarrow$ 87.8 (−2.1)	(+0.3)
33%	43.4 $\rightarrow$ 43.8 (+0.4)	69.8 $\rightarrow$ 67.9 (−1.9)	(−0.2)
16%	39.1 $\rightarrow$ 39.5 (+0.4)	68.9 $\rightarrow$ 81.8 (+12.9)	(−2.0)
10%	35.6 $\rightarrow$ 37.6 (+2.0)	36.5 $\rightarrow$ 67.4 (+30.9)	(+1.1)
5%	29.3 $\rightarrow$ 34.2 (+4.9)	18.9 $\rightarrow$ 44.8 (+25.9)	(−0.3)
2%	20.4 $\rightarrow$ 29.8 (+9.4)	16.3 $\rightarrow$ 30.0 (+13.7)	(+22.9)
1%	13.6 $\rightarrow$ 21.2 (+7.6)	4.0 $\rightarrow$ 19.8 (+15.8)	(+22.7)

each of the datasets used in previous experiments, past the point of saturation on the validation set, and save a model checkpoint once every 10 epochs. We then perform the SSOD training stage starting from each of those checkpoints. We use those results to try to find Spearman’s ρ rank correlation coefficient between the burn-in performance of a given checkpoint and its subsequent SSOD performance. Because we only have 30 pairs of burn-in and SSOD performance results for each dataset to determine the correlation, we perform bootstrapping with 10,000 samples to determine a 95% confidence interval for Spearman’s ρ .

In absence of an a-priori predictive metric, we additionally test a model soup²⁷ of all saved checkpoints of the extended burn-in as an SSOD starting point. A model soup is simply the average of those weights, similar to an EMA. While it is not likely to meet or exceed the highest performing individual checkpoint, it can significantly mitigate the effect of outliers in the opposite direction, offering a strong starting point without having to perform a large number training runs.

4. RESULTS

4.1 Artificially Constrained Labeled Data

Table 3 summarizes the results of the artificially constrained labeled data experiment on the VisDrone2019-DET, ATR, and TNO-A2G datasets. A consistent trend is observed across all three datasets: SSOD significantly improves the burn-in mAP50, particularly at lower percentages of labeled data (typically starting from 16%). The maximum mAP@50 improvements achieved are +9.4, +30.9, and +22.9 for VisDrone2019-DET, ATR, and TNO-A2G respectively.

Another notable pattern is that the performance gains are more pronounced when the proportion of labeled data is smaller, especially for ATR and TNO-A2G. These datasets contain relatively few images, so incorporating additional, diverse unlabeled data provides substantial benefits. However, adding a large amount of additional data beyond a certain point does not yield further improvements. This is likely because the extra data does not introduce new variations since most of it is sampled from videos, which limits diversity.

Table 4 shows that for ATR, the foundation model (OWL-ViT v2) outperforms the YOLOv5 burn-in model at lower labeled data percentages (up to 33%), likely due to its strong pretrained weights. As the amount of labeled data increases, the gap narrows, and when SSOD is applied, YOLOv5 consistently surpasses the foundation model, highlighting the effectiveness of leveraging unlabeled data. For VisDrone2019-DET, OWL-ViT v2 only competes in the lowest data regime, which can be attributed to the larger dataset size and the absence of advanced augmentations in our OWL-ViT fine-tuning pipeline, whereas YOLOv5 benefits from such augmentations.

Table 4: Results of the artificially constrained labeled data experiments with the foundation model, showing the foundation model performance (SSOD performance) in mAP@50 (and percentage of the SSOD it outperforms).

Labeled	VisDrone2019-DET	ATR
100%	32.6 (49.6) (+17.9)	88.2 (92.4) (+4.2)
66%	30.6 (46.9) (+16.5)	89.2 (87.8) (−1.4)
33%	30.4 (43.8) (+13.4)	84.2 (67.9) (−16.3)
16%	29.2 (39.5) (+10.3)	71.8 (81.8) (+10.0)
10%	27.0 (37.6) (+10.6)	66.6 (67.4) (+0.8)
5%	26.5 (34.2) (+7.7)	38.3 (44.8) (+6.5)
2%	24.8 (29.8) (+5.0)	28.5 (30.0) (+1.5)
1%	22.4 (21.2) (−1.2)	11.7 (19.8) (+8.1)



Figure 7: Some qualitative examples of predictions of the best burn-in model (i.e., without performing SSOD, top) and the best SSOD model (bottom). The bulk of the improvement for SSOD seems to come from the removal of confident false-positive predictions on background areas, most commonly square buildings or relatively high-contrast foliage. SSOD also succeeds in correctly detecting more objects that were previously missed.

4.2 UA-WAR Experiment

Table 5 shows the results of the UA-WAR experiment, where SSOD manages to improve from a top burn-in mAP50 of 41.1 to 44.8 after performing SSOD on the additional unlabeled images. The OWL-ViT v2 foundation model achieved an mAP@50 of 43.11, placing it between the best burn-in and SSOD results. Figure 7 shows a few qualitative examples of the most common improvements. The bulk of the improvement for SSOD seems to come from the removal of confident false-positive predictions on background areas, most commonly square buildings or relatively high-contrast foliage. SSOD also succeeds in correctly detecting more objects that were previously missed.

4.3 Burn-In and Model Soups

Table 5 shows the results of the burn-in stopping point and model soups experiments for all three datasets. From the large range of Spearman’s ρ confidence interval, the correlation being positive or negative depending on the choice of dataset, and the fact that in all cases the checkpoint with the best burn-in performance has a

Table 5: Results of the burn-in stopping point experiments, showing the burn-in $\rightarrow$ SSOD performance of the checkpoint with the highest burn-in performance (and percentage of checkpoints it outperforms), the best burn-in $\rightarrow$ SSOD performance among all 30 individual checkpoints, the model soup performance (and percentage of checkpoints it outperforms), and Spearman’s ρ and its 95% confidence interval indicating the rank correlation between the burn-in performance and the subsequent SSOD performance.

Dataset	Best Burn-In	Best SSOD	Model Soup	Burn vs SSOD ρ
VisDrone (5%)	30.4 $\rightarrow$ 33.3 (>30.4%)	26.0 $\rightarrow$ 34.2	36.1 (>100%)	-0.630 [-0.911, -0.171]
ATR (8%)	40.1 $\rightarrow$ 42.3 (>2.7%)	12.0 $\rightarrow$ 70.1	65.0 (>67.6%)	-0.952 [-0.977, -0.885]
UA-WAR	41.1 $\rightarrow$ 41.8 (>34.4%)	39.8 $\rightarrow$ 44.8	43.5 (>82.8%)	0.386 [-0.012, 0.711]

relatively low SSOD performance, are all strong indicators that burn-in performance is not predictive of SSOD performance. The model soup runs, however, consistently perform much better than the average individual checkpoint, making it a safe starting point when not able to train tens of different models to find the optimal starting point.

5. CONCLUSION

We have investigated the utility of SSOD on challenging, military-relevant datasets from a drone viewpoint, showing large potential for performance improvements especially at small amounts of labeled data. We identified a large variance in SSOD performance depending on the point at which one chooses to move from the burn-in to the SSOD phase, and demonstrated that burn-in performance is not predictive of SSOD performance. However, we show that model soups are a reliable approach to reach near-optimal performance without having to perform multiple SSOD stages from different starting points. We applied the lessons learned to a practical use case of videos from the frontline in Ukraine, showing that starting with just a small amount of labeled data and SSOD can result in a strong detector, which can in turn be used to bootstrap annotation efforts of more data. Finally, comparing SSOD with a foundation model (OWL-ViT v2) shows that the foundation model can be competitive in extreme low-label regimes (notably on ATR), yet is generally overtaken by YOLOv5 once SSOD is applied. This emphasizes the practical advantage of incorporating unlabeled data through SSOD, even when strong pretrained models are available.

ACKNOWLEDGMENTS

This work received funding from the European Defence Fund through the project STORE (Shared daTabase for Optronics image Recognition and Evaluation), grant agreement № 101121405.

REFERENCES

- [1] Aljawaheri, D., Burghouts, G. J., Van Woerden, J. E., Dijk, J., and Kuijf, H. J., “Zero-shot neuro-symbolic parsing of body keypoints,” in *[Electro-Optical and Infrared Systems: Technology and Applications XXI]*, Hickman, D. L., Bürsing, H., Soan, P. J., and Steinvall, O., eds., 48, SPIE, Edinburgh, United Kingdom (Nov. 2024).
- [2] Van Woerden, J. E., Burghouts, G. J., Van Rooij, S. B., Ruis, F. A., Dijk, J., and Kuijf, H. J., “Visual prompt tuning and ensemble undersampling for one-shot vehicle classification,” in *[Artificial Intelligence for Security and Defence Applications II]*, Bouma, H., Yitzhaky, Y., Prabhu, R., and Kuijf, H. J., eds., 16, SPIE, Edinburgh, United Kingdom (Nov. 2024).
- [3] The Economist, “Russian tanks in ukraine are sprouting cages,” (2022). Accessed: 2025-08-04.
- [4] Epstein, J., “Us-made abrams tanks are fighting a dangerous new kind of war, but ukrainians are building unusual armor to help them survive it,” (October 2024). Accessed: 2025-08-04.
- [5] Parken, O., “New russian turtle tank with cage-like armor emerges on ukrainian battlefield,” (May 2024). Accessed: 2025-08-04.

- [6] Business Insider, “Russia’s throwing away an advantage of the soviet-style tank by building weird cage,” (August 2024). Accessed: 2025-08-04.
- [7] Newdick, T., “M2 bradley fighting vehicle appears in russia with locally made 30mm cannon,” (June 2025). Accessed: 2025-08-04.
- [8] Sutton, H. I., “World first: Ukraine’s newest naval drone in unique attack on russian positions,” (December 2024). Accessed: 2025-08-04.
- [9] Altman, H., “New armor kits being installed on ukraine’s western air defense systems,” (January 2025). Accessed: 2025-08-04.
- [10] Xu, B., Chen, M., Guan, W., and Hu, L., “Efficient teacher: Semi-supervised object detection for yolov5,” (2023).
- [11] Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., and Dollár, P., “Microsoft coco: Common objects in context,” (2015).
- [12] Everingham, M., Gool, L. V., Williams, C. K. I., Winn, J. M., and Zisserman, A., “The pascal visual object classes (voc) challenge,” *Int. J. Comput. Vis.* **88**(2), 303–338 (2010).
- [13] Tarvainen, A. and Valpola, H., “Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results,” (2018).
- [14] Shehzadi, T., Ifza, Stricker, D., and Afzal, M. Z., “Semi-supervised object detection: A survey on progress from cnn to transformer,” (2024).
- [15] Jocher, G., Stoken, A., Borovec, J., NanoCode012, ChristopherSTAN, Changyu, L., Laughing, tkianai, Hogan, A., lorenzomammanna, yxNONG, AlexWang1900, Diaconu, L., Marc, wanghaoyang0106, ml5ah, Doug, Ingham, F., Frederik, Guilhen, Hatovix, Poznanski, J., Fang, J., , L. Y., changyu98, Wang, M., Gupta, N., Akhtar, O., PetrDvoracek, and Rai, P., “ultralytics/yolov5: v3.1 - bug fixes and performance improvements,” (Oct. 2020).
- [16] Ren, S., He, K., Girshick, R., and Sun, J., “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” (Jan. 2016). arXiv:1506.01497 [cs].
- [17] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S., “End-to-end object detection with transformers,” (2020).
- [18] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A., “Emerging properties in self-supervised vision transformers,” (2021).
- [19] Liu, Y.-C., Ma, C.-Y., He, Z., Kuo, C.-W., Chen, K., Zhang, P., Wu, B., Kira, Z., and Vajda, P., “Unbiased teacher for semi-supervised object detection,” (2021).
- [20] Xu, M., Zhang, Z., Hu, H., Wang, J., Wang, L., Wei, F., Bai, X., and Liu, Z., “End-to-end semi-supervised object detection with soft teacher,” (2021).
- [21] Zhang, J., Lin, X., Zhang, W., Wang, K., Tan, X., Han, J., Ding, E., Wang, J., and Li, G., “Semi-detr: Semi-supervised object detection with detection transformers,” (2023).
- [22] Minderer, M., Gritsenko, A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., Wang, X., Zhai, X., Kipf, T., and Houlsby, N., “Simple open-vocabulary object detection with vision transformers,” (2022).
- [23] Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., and Ling, H., “Detection and tracking meet drones challenge,” (2021).
- [24] Defense Systems Information Analysis Center (DSIAC), “ATR Algorithm Development Image Database.” <https://dsiac.org/databases/atr-algorithm-development-image-database/> (n.d.).
- [25] Van Lier, M. L., Den Hollander, R. J. M., and Kuijff, H. J., “Adversarial AI image perturbation attack invariant to object scale and type,” in [*Artificial Intelligence for Security and Defence Applications II*], Bouma, H., Yitzhaky, Y., Prabhu, R., and Kuijff, H. J., eds., 34, SPIE, Edinburgh, United Kingdom (Nov. 2024).
- [26] Minderer, M., Gritsenko, A., and Houlsby, N., “Scaling open-vocabulary object detection,” in [*Advances in Neural Information Processing Systems*], Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., eds., **36**, 72983–73007, Curran Associates, Inc. (2023).

- [27] Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., and Schmidt, L., “Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time,” in [*Proceedings of the 39th International Conference on Machine Learning*], Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S., eds., *Proceedings of Machine Learning Research* **162**, 23965–23998, PMLR (17–23 Jul 2022).