



# Self-Supervised Pretraining on Satellite Imagery: A Case Study on Label-Efficient Vehicle Detection

Jules Bourcier, Thomas Floquet, Gohar Dashyan, Tugdual Ceillier, Karteek Alahari, Jocelyn Chanussot

## ► To cite this version:

Jules Bourcier, Thomas Floquet, Gohar Dashyan, Tugdual Ceillier, Karteek Alahari, et al.. Self-Supervised Pretraining on Satellite Imagery: A Case Study on Label-Efficient Vehicle Detection. Conference on Artificial Intelligence for Defense, DGA Maîtrise de l'Information, Nov 2022, Rennes, France. hal-03881747

**HAL Id: hal-03881747**

**<https://hal.science/hal-03881747>**

Submitted on 2 Dec 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

# Self-Supervised Pretraining on Satellite Imagery: A Case Study on Label-Efficient Vehicle Detection

1<sup>st</sup> Jules Bourcier

*Preligens (ex-Earthcube)*  
*Inria, Univ. Grenoble Alpes,*  
*CNRS, Grenoble INP, LJK*  
jules.bourcier@preligens.com

2<sup>nd</sup> Thomas Floquet

*Preligens (ex-Earthcube)*  
*MINES Paris - PSL University*  
thomas.floquet@minesparis.psl.eu

3<sup>rd</sup> Gohar Dashyan

*Preligens (ex-Earthcube)*  
gohar.dashyan@preligens.com

4<sup>th</sup> Tugdual Ceillier

*Preligens (ex-Earthcube)*

5<sup>th</sup> Karteek Alahari

*Inria, Univ. Grenoble Alpes,*  
*CNRS, Grenoble INP, LJK*

6<sup>th</sup> Jocelyn Chanussot

*Inria, Univ. Grenoble Alpes,*  
*CNRS, Grenoble INP, LJK*

**Abstract**—In defense-related remote sensing applications, such as vehicle detection on satellite imagery, supervised learning requires a huge number of labeled examples to reach operational performances. Such data are challenging to obtain as it requires military experts, and some observables are intrinsically rare. This limited labeling capability, as well as the large number of unlabeled images available due to the growing number of sensors, make object detection on remote sensing imagery highly relevant for self-supervised learning. We study in-domain self-supervised representation learning for object detection on very high resolution optical satellite imagery, that is yet poorly explored. For the first time to our knowledge, we study the problem of label efficiency on this task. We use the large land use classification dataset Functional Map of the World to pretrain representations with an extension of the Momentum Contrast framework. We then investigate this model’s transferability on a real-world task of fine-grained vehicle detection and classification on Preligens proprietary data, which is designed to be representative of an operational use case of strategic site surveillance. We show that our in-domain self-supervised learning model is competitive with ImageNet pretraining, and outperforms it in the low-label regime.

**Index Terms**—deep learning, computer vision, remote sensing, self-supervised learning, object detection, land use classification, label-efficient learning

## I. INTRODUCTION

Very high resolution (VHR) satellite imagery is one of the key data from which geospatial intelligence can be gathered. It is an essential tool to detect and identify a wide range of objects, on very large areas and on a very frequent basis. Recently, we have seen the multiplication of available sensors, which has led to a large increase in the volume of data available. This makes it very challenging for human analysts to exploit these data without resorting to automatic solutions. Deep learning techniques today have been highly effective to perform such tasks. However, training those models requires very large labeled datasets. Annotating objects of interest in VHR images can prove to be very costly, being both difficult and time-consuming, and requiring fine domain expertise. In specific contexts such as in geospatial intelligence, the targets can be intrinsically rare, difficult to localize and to identify accurately. This makes the acquisition of thousands of examples impractical, as is typically required for classic supervised deep learning methods to generalize. Consequently, a major challenge is the development of label-efficient approaches, i.e., models that learn with few annotated examples.

To reduce the number of training samples for

difficult vision tasks such as object detection, transfer learning of pretrained neural networks is used extensively. The idea is to reuse a network trained *upstream* on a large, diverse source dataset. ImageNet [19] has become the de facto standard for pretraining: due to its large-scale and genericity, ImageNet-pretrained models show to be adaptable beyond their source domain, including remote sensing imagery [17]. Nonetheless, the *domain gap* between ImageNet and remote sensing domains brings questions about the limitations of this transfer when there are very few samples on the task at hand, e.g., the detection of rare observables from satellite images. To fit the distributions of *downstream* tasks with maximum efficiency, one would ideally use generic *in-domain* representations, obtained by pre-training on large amounts of remote sensing data. This is infeasible in the remote sensing domain due to the difficulty of curating and labeling these data at the scale of ImageNet. However, imaging satellites provide an ever-growing amount of unlabeled data, which makes it highly relevant for learning visual representations in an unsupervised way.

Self-supervised learning (SSL) has recently emerged as an effective paradigm for learning representations on unlabeled data. It uses unlabeled data as a supervision signal, by solving a *pretext task* on these input data, in order to learn semantic representations. A model trained in a self-supervised fashion can then be transferred using the same methods as a network pretrained via an upstream supervised task. In the last two years, SSL has shown impressive results that closed the gap or even outperformed supervised learning for multiple benchmarks [3], [4], [8], [9]. Recently, SSL has been applied in the remote sensing domain to exploit readily-available unlabeled data, and was shown to reduce or even close the gap with transfer from ImageNet [1], [16], [26]. Nonetheless, the capacity of these methods to generalize from few labels has not been explored on the important problem of object detection in VHR satellite images.

In this paper, we explore in-domain self-supervised representation learning for the task of object detection on VHR optical satellite imagery. We use the large land use classification dataset

Functional Map of the World (fMoW) [6] to pretrain representations using the unsupervised framework of MoCo [9]. We then investigate the transferability on a difficult real-world task of fine-grained vehicle detection on proprietary data, which is designed to be representative of an operational use case of strategic site surveillance. Our contributions are:

- We apply a method based on MoCo with temporal positives [1] to learn self-supervised representations of remote sensing images, that we improve using (i) additional augmentations for rotational invariance; (ii) a fixed loss function that removes the false temporal negatives in the learning process.
- We investigate the benefit of in-domain self-supervised pretraining as a function of the annotation effort, using different budgets of annotated instances for detecting vehicles.
- We show that our method is better than or at least competitive with supervised ImageNet pretraining, despite using no upstream labels and 3× less upstream data.

Furthermore, our in-domain SSL model is more label-efficient than ImageNet: when using very limited annotations budgets ( $\simeq 20$  images totalling  $\simeq 12k$  observables), we outperform ImageNet pretraining by 4 points AP on vehicle detection and 0.5 point mAP on joint detection and classification.

## II. RELATED WORK

### A. Self-supervised representation learning

SSL methods use unlabeled data to learn representations that are transferable to downstream tasks (e.g. image classification or object detection) for which annotated data samples are insufficient. In recent years, these methods have been successfully applied to computer vision with impressive results that closed the gap or even outperformed supervised representation learning on ImageNet, on multiple benchmarks including classification, segmentation, and object detection [3], [4], [8], [9]. They commonly rely on a pretraining phase, where a neural network, a *representation encoder*, is trained to solve a pretext task, for which generating labels does not require any effort or human involvement. Solving the pretext task is done only for the true

purpose of learning good data representations that allow for efficient training on a downstream task of genuine interest.

### B. Contrastive learning

*Contrastive learning* has recently become the most competitive unsupervised representation learning framework, with approaches such as MoCo [9], SimCLR [4], and SwAV [3]. Contrastive methods work by attracting embeddings of pairs of samples known to be semantically similar (*positive pairs*) while simultaneously repelling pairs of unlike samples (*negative pairs*). The most common way to define similarity is to use the *instance discrimination* pretext task [7], [24], in which positives are generated as random data augmentations on the same image, and negatives are simply generated from different images. Thanks to this pretext task, the encoder learns similar representations for several views of the same object instance in an image and distant representations for different instances. *Momentum Contrast* (MoCo) [9] is a strong contrastive method that implements a dynamic dictionary with a queue and a moving-averaged encoder, which enables building a large and consistent dictionary on-the-fly (see section III-A for more details). In this paper, we adopt a recent *geography-aware* rework of this approach made by [1].

### C. Representation learning in remote sensing

Building on that success in computer vision, SSL has recently been applied to remote sensing and was shown to reduce or even close the gap with transfer from ImageNet. [12] first made use of contrastive learning for remote sensing representation learning and [13] also apply a spatial augmentation criteria on top of MoCo [9]. These works exploit relevant prior knowledge about the remote sensing domain: the assumption that images that are geographically close should be semantically more similar than distant images. Another way of making the learning procedure *geography-aware* is to exploit the image time series that one can get from a given geographic area thanks to the frequent revisit of satellites. This approach is adopted by [1], that use spatially aligned images

over time to construct *temporal positive pairs*. With their *geography-aware* representations learned on the fMoW dataset [6], they improve significantly on classification, segmentation and object detection downstream tasks. However, they do not study label efficiency. In this paper, we apply this method, called MoCoTP, to an operational use-case. We study the label efficiency and bring small extensions to this model, that further improve its performance. In the same vein, [16] present a pipeline for self-supervised pretraining on uncurated remote sensing data and propose a method that learns representations that are simultaneously variant and invariant to temporal changes. They significantly outperform ImageNet pretraining on classification from few labels on medium resolution images (10m). One can also exploit the multispectral and multisensor nature of remote sensing. [20] split multispectral images into two different subsets of channels and use them as augmented (positive) views. [21] extend this to multiple sensors, taking subsets of the combination of all bands. Regarding the domain of pretraining data in remote sensing, [17] show that the relatedness of the pretraining distribution to the downstream task can improve performances in low-labeled settings. However, they only study supervised pretraining and classification tasks, and show that transfer performance is very dependent on the labeling and data curation quality in the pretraining dataset. Therefore this leaves unresolved the problem of obtaining generic representations with less dependence on labels and this is where SSL can help.

## III. METHOD

In this section, we detail our approach to explore the applicability of SSL to vehicle detection and classification on optical satellite imagery. The overall procedure is described in Fig. 1. We first pretrain a ResNet-50 backbone on the fMoW dataset [6] in an unsupervised way with a recent SSL method, MoCoTP [1]. See section III-A for details on the approach and section IV-A2 for implementation details.

We use the weights of this pretrained backbone on two downstream tasks: (i) fMoW image recog-

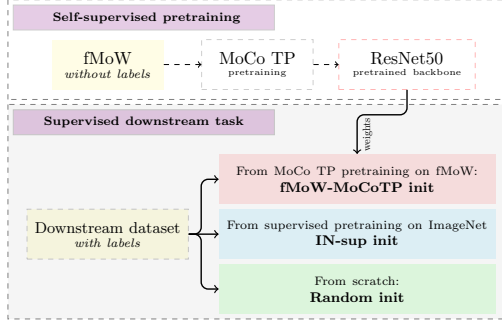


Fig. 1. Schematic outline of our method. The top block represents the pretraining phase. The bottom block represents the pretrained weights that are injected into the downstream task model.

dition task: we inject the weights in a classifier, and perform *linear probing* and *finetuning*. See section IV-A for more details. (ii) Vehicle detection and classification on Preligens proprietary data: we inject the weights in a RetinaNet detector [15], and finetune the entire model. See section IV-B2 for implementation details.

#### A. Self-supervised learning with MoCo and Temporal Positives

We employ the MoCo [9] framework for contrastive SSL. The base method we use is from the improved variant MoCo-V2 [5]. MoCo learns to match an input *query*  $q$  to a *key*  $k^+$  (representing the encoded views of the same sample) among a set of negative keys  $k^-$ , using the instance discrimination pretext task [24]. It uses a deep encoder (e.g. a ResNet [10]) to map input image queries and keys to a vector representation space. Negative keys are extracted with a moving average network (momentum encoder) to maintain consistent representations during training, and are drawn from a memory queue. We refer readers to [9] for details on this. MoCo uses the popular choice of InfoNCE [18] for the contrastive loss:

$$\mathcal{L}(q, k^+) = -\log \frac{e^{(q \cdot k^+ / \tau)}}{e^{(q \cdot k^+ / \tau)} + \sum_{k^-} e^{(q \cdot k^- / \tau)}} \quad (1)$$

where  $\tau$  is a temperature scaling parameter.

On top of MoCo, we adopt the extension to temporal views proposed in [1], MoCo with Tem-

poral Positives (MoCoTP). It extends the instance discrimination pretext task to use spatially aligned images from different times as positives. Maximizing similarity between temporal views can provide richer semantic information that extracts persistent scene features over time [1]. The same random augmentations as in MoCo-V2 are also applied on the temporal samples.

**Improvements to MoCoTP.** Compared to MoCoTP, we adopt the following modifications from [2], to improve the framework of [1]: (i) In addition to the geometric and color perturbations of MoCo-V2, we apply random vertical flips and rotations by multiples of  $90^\circ$ . Since the data augmentation scheme plays a leading role in contrastive learning [22], we aim to learn representations more suited to overhead images thanks to rotational invariance. (ii) [1] introduces temporal positives as a drop-in replacement for  $q$  and  $k^+$  in (1). However, this can introduce *false negatives*. Indeed, at each iteration of training, it may happen that the set of negatives  $k^-$  contains temporal views for samples of the current mini-batch of queries. Such false negatives will cause an incorrect repulsion between the embeddings of similar samples. To avoid the false negatives to interfere with the learning objective, we simply mask out the logits  $q \cdot k^-$  in the InfoNCE loss in (1) for every  $k^-$  that happens to be a temporal view of  $q$ .

After being trained on the pretext task for a given number of iterations, the query encoder is extracted and can be transferred to downstream tasks.

#### B. Transfer to downstream task

Following the MoCoTP pretraining on fMoW, we transfer the obtained weights on two downstream tasks: fMoW image recognition task and a real-world use-case, vehicle detection and classification on Preligens proprietary data. We refer to the downstream training initialized with SSL weights learned on the fMoW dataset as **fMoW-MoCoTP init**. For each downstream task, we compare fMoW-MoCoTP init with two baselines:

- **IN-sup init**: the backbone has been pretrained on ImageNet in a supervised way;

- **Random init:** the backbone is initialized randomly (i.e., no pretraining).

#### IV. EXPERIMENTAL SETUP

##### A. Pretraining and evaluation on fMoW

1) *Dataset:* For the sake of learning semantic representations in remote sensing, we adopt the fMoW dataset [6], following [1]. fMoW is a public dataset of VHR imagery from Maxar Earth observation satellites, is large-scale with 363,571 training images, and covers 207 countries. It provides images from same locations over time. We apply the self-supervised MoCoTP method for pretraining on fMoW training set using these available temporal views. fMoW also includes ground-truths labels for functional land use classification with 62 diverse categories with a long-tailed distribution. We do not use those labels for self-supervised pretraining, but use them downstream to evaluate representations learned directly for the classification of the images seen during pretraining. Following [6] and [1], we use the fMoW-RGB products for our experiments, which provides 3-bands imagery at 0.5m ground resolution. Preprocessing is applied identically to [6] to resize input images to 224×224 pixels.

Ensuing the pretraining stage, the model is evaluated on the land use classification task with the two protocols of linear probing (training a linear classifier on top of frozen features from the pre-trained encoder) or finetuning (updating all parameters of the network). To study the label efficiency of the learned representations, supervised evaluation is performed on 1%, 10%, and 100% of the labeled training data. For 1% and 10% labels, we subsample by preserving the base class distribution. The evaluation metric is the F1-score averaged over classes. Testing is performed on the fMoW validation set for comparison with [1], which consists of 53,041 images.

2) *Implementation details:* In all our experiments, we use MoCoTP with ResNet-50 for the query and key encoders. We utilize the same hyperparameters as in [2]. For self-supervised pretraining, we use a learning rate of 3e-2 with a cosine schedule, batch size of 256, dictionary queue size of 65536, temperature scaling of 0.2, SGD optimizer

with a momentum of 0.9, weight decay of 1e-4. Augmentations are the same as in MoCo-V2, plus a vertical flip with 0.5 probability and a 90° rotation with 0.75 probability. Pretraining lasts 200 epochs. For linear probing, we use a learning rate of 1, no weight decay, and only random resized cropping for the augmentations. For finetuning, we use a learning rate of 3e-4 for ResNet weights and 1 for the final classification layer, weight decay of 1e-4, and the same augmentations used for pretraining. We compare self-supervised pretraining against Random init and IN-sup init, under the different label regimes. Models are trained with cross-entropy loss and evaluated on epoch with the highest top-1 accuracy on the validation set.

##### B. Transfer to vehicle instance detection

1) *Dataset:* We describe here the Preligens proprietary datasets used for the transfer to object detection. Statistics of our datasets are reported in Table I.

We call ‘S’ our base dataset. It consists of 204 Maxar WorldView-3 satellite images at 0.3m resolution, and approximately 150k vehicles. To study the label efficiency, we subsample this base dataset S into smaller datasets, ‘XS’ and ‘XXS’, targeting respectively 50% and 10% of the observables present in S. We ensure that XXS is included in XS, so that our datasets follow a “Matriochka” structure, which simulates the incremental nature of annotation efforts. The sampling strategy is such that the class distribution is well preserved. To perform variance experiments on our results, we run the sampling thrice and get three different variants of the XS and XXS datasets. We proceed similarly for the S training set by selecting other satellite images that match the class distribution of the initial S dataset. We keep the same validation and testing sets throughout the experiments, and make sure that the geographical sites of the images are distinct between train and test splits. The training and validation raster images are divided into tiles of 512x512 pixels with an overlap of 128 pixels. We take all positive tiles (i.e., tiles containing at least one instance) and some negative tiles, with a positive/negative ratio of 5 and 3 for training and

validation sets respectively: this setting allows to focus training efforts on positive tiles while keeping a fair amount of negative tiles. The ground truth labels are non-oriented bounding boxes of target observables with their class label. Our classification problem is composed of 8 vehicle categories: *Civilian*, *Military*, *Armored*, *Launcher*, *Ground Support Equipment (GSE)*, *Electronics*, *Heavy Equipment (HE)*, and *Lifting Equipment (LE)*. The classes are naturally heavily imbalanced: the first three classes are very dominant, covering  $\sim 96.5\%$  of the vehicles in our datasets, while the last five classes are very under-represented.

## 2) Implementation details:

The detection model. The object detection model used is a RetinaNet [15], with a ResNet-50-FPN backbone [14] with pyramid levels  $P_2$  to  $P_6$ . The backbone is initialized with the pretrained weights, while the detector specific modules are initialized randomly. We finetune the RetinaNet model end-to-end, and use the focal loss objective for the classification [15].

Hyperparameters. We select a learning rate of  $1e-4$ , an Adam optimizer and a batch size of 8, and no weight decay. As data augmentations, we use  $90^\circ$  rotations and flips, as well as CLAHE. We train until convergence and reduce the learning rate by 2 on plateau of the validation loss. Then, the epoch selected for the evaluation of the model is the one achieving the best F1-score calculated on the validation set, with a fixed detection score threshold of 0.15.

Evaluation. The F1-score is computed from the precision-recall curves obtained by varying the detection threshold from 0.15 to 0.9. In the following, we refer to the results on the task of vehicle detection (regardless of the class) as *level-1* results, and on the task of joint detection and classification as *level-2* results. The IoU threshold matching the predictions with the ground truths is set to 0.0, which is operationally relevant for observable counting purposes. In addition to the level-1 F1-score, we also compute the level-1 average precision (AP) and level-2 mean average precision (mAP) which are commonly used metrics to evaluate detection models.

## V. RESULTS

### A. fMoW classification

Table II shows the results of linear probing and finetuning on the 62-class land use classification task of fMoW. With 100% labels, we see that our improved reproduction of MoCoTP increases performance by 4.36 pts compared to [1] in linear probing and 1.62 pts in finetuning. This shows that the use of the rotation augmentations and the correction of false negatives in the loss function is helpful, especially for linear probing, which closes the gap with finetuning completely. As a note, the sole additional augmentations also improve the baselines Random init and IN-sup init of [1] by 1.29 pts and 0.68 pts respectively. Moreover, MoCoTP shows impressive label efficiency: in the semi-supervised settings of 1% and 10% labels, we see that it gives respectively 96% and 87% of the performance of the network trained with 100% labels. Also, it surpasses IN-sup init by large margins, with e.g. +20.57 pts on 1% finetune. These results indicate that MoCoTP is very efficient at learning semantic features from the upstream dataset. Therefore, this is encouraging in order to transfer to a downstream operational task where labeled data are scarce.

### B. Transfer to vehicle detection

1) *Label efficiency:* Table III and Fig. 2 show the results on vehicle detection. The F1-score with fMoW-MoCoTP init is always higher than with IN-sup init or Random init. fMoW-MoCoTP init achieves an F1-score of 65.1% with only 12k observables on dataset XXS. On dataset XS, with 50% less examples than on dataset S, fMoW-MoCoTP init is only 3.8 pts below the score obtained on dataset S. Moreover, the smaller the dataset, the larger the gap between fMoW-MoCoTP init and IN-sup init or Random init: on dataset S, fMoW-MoCoTP init's F1-score is 5.20 pts better than Random init on average, and also 0.40 pts better than IN-sup init, whereas on the XXS dataset, fMoW-MoCoTP init's F1-score is 39 pts better than Random init, and 3.7 pts better than IN-sup init. These results show that self-supervised in-domain pretraining can be competitive with supervised pretraining on ImageNet, and even give better results

TABLE I  
DATA STATISTICS FOR VEHICLE TRAINING, VALIDATION AND TESTING SETS. FOR EACH OF THE XXS, XS AND S TRAINING SETS, THE REPORTED NUMBERS ARE THE MEAN NUMBER OF OBSERVABLES BETWEEN THE DIFFERENT SAMPLED SETS.

| Dataset | Images | Pos. tiles | Neg. tiles | Vehicles | Civilian | Military | Armored | GSE | Launcher | Electronics | HE  | LE  |
|---------|--------|------------|------------|----------|----------|----------|---------|-----|----------|-------------|-----|-----|
| XXS     | 19     | 597        | 113        | 11,833   | 6,668    | 3,215    | 1,516   | 154 | 158      | 64          | 33  | 23  |
| XS      | 108    | 3,152      | 599        | 58,189   | 32,937   | 14,719   | 8,466   | 571 | 732      | 385         | 237 | 139 |
| S       | 204    | 6,438      | 1,231      | 115,617  | 66,504   | 29,332   | 16,148  | 820 | 1364     | 698         | 432 | 319 |
| Val     | 63     | 2,526      | 4,178      | 53,204   | 31,339   | 11,919   | 8,464   | 607 | 412      | 270         | 130 | 101 |
| Test    | 88     | –          | –          | 32,550   | 19,923   | 7,542    | 3,872   | 361 | 334      | 237         | 184 | 97  |

TABLE II  
RESULTS ON fMoW CLASSIFICATION (F1-SCORE IN %). FOR 1% AND 10% LABELS, THE VALUES ARE 'MEAN (SD)' ACROSS 3 REPLICATES WITH VARYING TRAINING SAMPLES. \* DENOTE OUR IMPROVED REPRODUCTIONS AS DETAILED IN SECTION III-A

|                      | 1% labels           |                     | 10% labels          |                     | 100% labels  |              |
|----------------------|---------------------|---------------------|---------------------|---------------------|--------------|--------------|
|                      | Linear              | Finetune            | Linear              | Finetune            | Linear       | Finetune     |
| Random init [1]      | –                   | –                   | –                   | –                   | –            | 64.71        |
| IN-sup init [1]      | –                   | –                   | –                   | –                   | –            | 64.72        |
| fMoW-MoCoTP init [1] | –                   | –                   | –                   | –                   | 64.53        | 67.34        |
| Random init *        | –                   | 19.29 (1.65)        | –                   | 51.87 (0.50)        | –            | 65.39        |
| IN-sup init *        | 32.41 (0.17)        | 39.43 (1.53)        | 43.86 (0.07)        | 57.32 (0.07)        | 50.25        | 66.01        |
| fMoW-MoCoTP init *   | <b>60.05 (0.11)</b> | <b>60.00 (0.43)</b> | <b>66.15 (0.11)</b> | <b>66.35 (0.75)</b> | <b>68.89</b> | <b>68.96</b> |

TABLE III  
RESULTS OF EACH METHOD ON EACH DATASET FOR VEHICLE DETECTION (%). THE VALUES ARE 'MEAN (SD)' ACROSS 3 REPLICATES WITH VARYING TRAINING SAMPLES.

| Metric           | F1                |                   |                   | AP                |                   |                   | mAP              |                   |                   |
|------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|-------------------|-------------------|
|                  | XXS               | XS                | S                 | XXS               | XS                | S                 | XXS              | XS                | S                 |
| Training set     |                   |                   |                   |                   |                   |                   |                  |                   |                   |
| Random init      | 26.0 (1.9)        | 55.1 (2.2)        | 74.7 (1.4)        | 12.3 (1.9)        | 46.9 (2.4)        | 75.3 (3.4)        | 2.2 (0.3)        | 8.4 (0.6)         | 16.3 (0.5)        |
| IN-sup init      | 61.4 (0.5)        | 75.1 (0.8)        | <b>79.5 (0.6)</b> | 56.1 (0.9)        | 75.6 (0.5)        | <b>80.9 (0.9)</b> | 9.3 (0.4)        | <b>14.9 (0.6)</b> | <b>19.7 (1.1)</b> |
| fMoW-MoCoTP init | <b>65.1 (0.8)</b> | <b>76.1 (0.4)</b> | <b>79.9 (0.3)</b> | <b>60.1 (1.6)</b> | <b>77.3 (0.1)</b> | <b>81.6 (0.5)</b> | <b>9.7 (0.2)</b> | <b>14.4 (0.4)</b> | <b>19.3 (1.0)</b> |

in low-label regimes. Moreover, Fig. 3 shows an example of predictions for models finetuned on an XS dataset, that illustrates qualitative improvements of fMoW-MoCoTP against IN-sup init on both false positive and false negative detections.

2) *Dominant vs. rare classes*: Table IV shows the APs in level-2. fMoW-MoCoTP init achieves significantly higher results than Random init on these six classes. Also, fMoW-MoCoTP init achieves higher AP than IN-sup init on the three dominant classes (*Civilian*, *Military* and *Armored*), that account for  $\sim 96.5\%$  of the vehicles in our datasets. However, IN-sup init outperforms fMoW-MoCoTP init on the very rare *Launcher*, *Electronics* and *HE* classes, that cover  $\sim 2.2\%$  of the vehicles

in our datasets. Since mAP gives equal importance to all classes, this translates into much closer values for mAP scores than level-1 AP scores between fMoW-MoCoTP init and IN-sup init methods, as one can see in Fig. 2. This might suggest that fMoW-MoCoTP init is *lazier* and mainly focuses on the dominant classes. The fMoW dataset used for pretraining contains a long-tailed distribution of semantic categories. One could hypothesize that this leads to representations being more skewed towards over-represented visual concepts than ImageNet, as the latter contains a balanced set of categories, and that such bias may also negatively impact the transfer to under-represented classes downstream. However, further work is needed to provide ground



TABLE IV

AP PER CLASS (%). RESULTS ON GSE AND LE ARE OMITTED BECAUSE THEY ARE CLOSE TO ZERO DUE TO THE POOR NUMBER OF EXAMPLES IN THE DATASETS. THE VALUES ARE 'MEAN (SD)' ACROSS 3 REPLICATES WITH VARYING TRAINING SAMPLES.

| Dominant classes | Civilian          |                   |                   | Military          |                   |                   | Armored          |                  |                    |
|------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|--------------------|
| Training set     | XXS               | XS                | S                 | XXS               | XS                | S                 | XXS              | XS               | S                  |
| Random init      | 14.4 (1.8)        | 48.1 (1.2)        | 75.4 (4.5)        | 3.1 (1.5)         | 14.8 (0.6)        | 29.1 (3.2)        | 0.0 (0.0)        | 3.9 (3.5)        | 14.9 (7.7)         |
| IN-sup init      | 54.2 (1.0)        | 75.4 (0.8)        | 81.9 (0.7)        | <b>17.2 (3.8)</b> | <b>25.0 (2.1)</b> | 32.8 (1.9)        | <b>0.8 (0.4)</b> | <b>5.7 (0.8)</b> | <b>21.4 (11.4)</b> |
| fMoW-MoCoTP init | <b>59.9 (1.0)</b> | <b>79.5 (0.8)</b> | <b>83.6 (1.1)</b> | <b>17.9 (0.9)</b> | <b>25.0 (1.4)</b> | <b>35.4 (3.0)</b> | 0.2 (0.1)        | <b>6.5 (1.6)</b> | <b>22.4 (11.3)</b> |

| Rare classes     | Launcher         |                  |                   | Electronics |                  |                  | HE        |                  |                  |
|------------------|------------------|------------------|-------------------|-------------|------------------|------------------|-----------|------------------|------------------|
| Training set     | XXS              | XS               | S                 | XXS         | XS               | S                | XXS       | XS               | S                |
| Random init      | 0.0 (0.0)        | 0.4 (0.3)        | 6.0 (2.8)         | 0.0 (0.0)   | 0.0 (0.0)        | 0.1 (0.1)        | 0.0 (0.0) | 0.0 (0.0)        | 0.0 (0.0)        |
| IN-sup init      | <b>0.3 (0.2)</b> | <b>9.4 (2.4)</b> | <b>18.8 (4.7)</b> | 0.0 (0.0)   | <b>1.5 (0.8)</b> | <b>5.0 (0.2)</b> | 0.0 (0.0) | <b>0.3 (0.5)</b> | <b>1.1 (1.0)</b> |
| fMoW-MoCoTP init | 0.1 (0.1)        | 4.1 (1.1)        | 13.0 (5.7)        | 0.0 (0.0)   | 0.4 (0.4)        | 1.6 (0.4)        | 0.0 (0.0) | 0.0 (0.0)        | 0.0 (0.1)        |

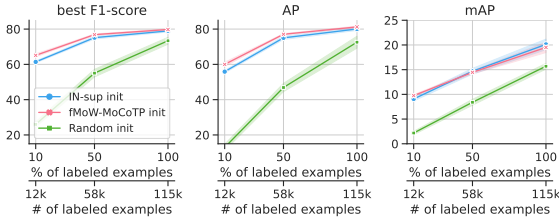


Fig. 2. Graphical view of the results of Table III.



Fig. 3. Example of cherry-picked predictions. Here the models have been finetuned on an XS dataset. White regions highlights errors specific to the top model, in solid and dotted style for false negatives and false positives respectively.

for this hypothesis.

## VI. CONCLUSION

In this work, we explored the added value of in-domain SSL for a real-world defense-related

remote sensing application: vehicle detection and classification on VHR optical satellite imagery. Considering this downstream task, we compared in-domain pretraining on the fMoW dataset with the traditional supervised ImageNet pretraining. We showed that self-supervised pretraining on fMoW is either competitive with or better than supervised ImageNet pretraining, despite using no upstream labels and 3× less upstream data. To study label efficiency, experiments were performed for different downstream dataset sizes, thereby mimicking different annotation budgets. We showed that in-domain SSL pretraining leads to better label efficiency than supervised pretraining on ImageNet. This result is particularly suitable for defense industry use cases, where labeling data is challenging. Further work could include additional studies such as increasing the amount of in-domain pretraining data, using a vehicle dataset that is more balanced in terms of classes (or deprived of its dominant classes), extending the range of sizes for the downstream dataset, and experimenting other SSL methods, including methods designed specifically for dense downstream tasks like detection [11], [23], [25].

## Acknowledgments

This work was performed using HPC resources from GENCI-IDRIS (Grant 2021-AD011013097). Karteek Alahari was supported in part by the ANR grant AVENUE (ANR-18-CE23-0011).

## REFERENCES

- [1] K. Ayush, B. Uz kent, C. Meng, K. Tanmay, M. Burke, D. Lobell, and S. Ermon, "Geography-aware self-supervised learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 181–10 190.
- [2] J. Bourcier, G. Dashyan, J. Chanussot, and K. Alahari, "Evaluating the label efficiency of contrastive self-supervised learning for multi-resolution satellite imagery," in *Image and Signal Processing for Remote Sensing XXVIII*, vol. 12267. SPIE, 2022, p. 122670K.
- [3] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9912–9924, 2020.
- [4] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.
- [5] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv preprint arXiv:2003.04297*, 2020.
- [6] G. Christie, N. Fendley, J. Wilson, and R. Mukherjee, "Functional map of the world," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6172–6180.
- [7] A. Dosovitskiy, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with convolutional neural networks," *Advances in neural information processing systems*, vol. 27, 2014.
- [8] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar *et al.*, "Bootstrap your own latent—a new approach to self-supervised learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 271–21 284, 2020.
- [9] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9729–9738.
- [10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [11] O. J. Hénaff, S. Koppula, J.-B. Alayrac, A. van den Oord, O. Vinyals, and J. Carreira, "Efficient visual pre-training with contrastive detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 086–10 096.
- [12] N. Jean, S. Wang, A. Samar, G. Azzari, D. Lobell, and S. Ermon, "Tile2vec: Unsupervised representation learning for spatially distributed data," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 3967–3974.
- [13] J. Kang, R. Fernandez-Beltran, P. Duan, S. Liu, and A. J. Plaza, "Deep unsupervised embedding for remotely sensed images based on spatially augmented momentum contrast," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 3, pp. 2598–2610, 2021.
- [14] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.
- [15] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.
- [16] O. Mañas, A. Lacoste, X. Giro-i Nieto, D. Vazquez, and P. Rodriguez, "Seasonal contrast: Unsupervised pre-training from uncured remote sensing data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9414–9423.
- [17] M. Neumann, A. S. Pinto, X. Zhai, and N. Houlsby, "In-domain representation learning for remote sensing," in *AI for Earth Sciences Workshop at International Conference on Learning Representations (ICLR)*, 2020, pp. 1–20.
- [18] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.
- [19] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.
- [20] V. Stojnic and V. Risojevic, "Self-supervised learning of remote sensing scene representations using contrastive multi-view coding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1182–1191.
- [21] A. M. Swope, X. H. Rudelis, and K. T. Story, "Representation learning for remote sensing: An unsupervised sensor fusion approach," *arXiv preprint arXiv:2108.05094*, 2021.
- [22] Y. Tian, C. Sun, B. Poole, D. Krishnan, C. Schmid, and P. Isola, "What makes for good views for contrastive learning?" *Advances in Neural Information Processing Systems*, vol. 33, pp. 6827–6839, 2020.
- [23] X. Wang, R. Zhang, C. Shen, T. Kong, and L. Li, "Dense contrastive learning for self-supervised visual pre-training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3024–3033.
- [24] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3733–3742.
- [25] Z. Xie, Y. Lin, Z. Zhang, Y. Cao, S. Lin, and H. Hu, "Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 684–16 693.
- [26] X. Zheng, B. Kellenberger, R. Gong, I. Hajnsek, and D. Tuia, "Self-supervised pretraining and controlled augmentation improve rare wildlife recognition in uav images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 732–741.