A REVIEW ON CONTRASTIVE LEARNING METHODS AND APPLICATIONS TO ROOF-TYPE CLASSIFICATION ON AERIAL IMAGES

Ahmed Ben Saad [1] *Sebastien Drouyer*[1] *Bastien Hell*[2]
Sylvain Gavoille [2] Stéphane Gaiffas[2,3] Gabriele Facciolo[1]

[1] Université Paris-Saclay, CNRS, ENS Paris-Saclay, Centre Borelli, France
[2] namR
[3] LPSM, Université de Paris, DMA, Ecole normale supérieure

ABSTRACT

Unsupervised learning based on Contrastive Learning (CL) has attracted a lot of interest recently. This is due to excellent results on a variety of subsequent tasks (especially classification) on benchmark datasets (ImageNet, CIFAR-10, etc.) without the need of large quantities of labeled samples. This work explores the application of some of the most relevant CL techniques on a large unlabeled dataset of aerial images of building rooftops. The task that we want to solve is roof type classification using a much smaller labeled dataset. The main problem with this task is the strong dataset bias and class imbalance. This is caused by the abundance of certain types of roofs and the rarity of other types. Quantitative results show that this issue heavily affects the quality of learned representations, depending on the chosen CL technique.

***Index Terms*—** Contrastive Learning, Aerial Imagery, Classification, Data Bias, Data Inbalance.

1. INTRODUCTION

Unsupervised contrastive learning is the subject of many research papers from last year. This is mainly due to recent breakthroughs in this area [1, 2, 3, 4]. It seeks to leverage the vast quantity of unlabeled data available in the wild in order to learn image representations that can be used efficiently in various tasks. It has been proven that this technique can be used successfully as a pretraining step, since it can learn features that maximize the mutual information between the input and the representations [1].

In this work, we apply some of these techniques to aerial images of building rooftops. Examples from our dataset are shown in Fig. 1 and our downstream task is the classification of buildings roof types. The use of contrastive learning for this task is justified by the abundance of unlabeled data available in the wild, while labeled examples are scarce due to the prohibitive cost of labeling. One of the main problems that we had to face is the fact that there is one predominant roof type in all our aerial images: this makes labeled images very imbalanced (see Fig. 2), but also causes a dataset bias in unlabeled images. In our experiments, we observed this problem to be a huge obstacle both for unsupervised pretraining and supervised fine-tuning: the resulting networks failed at distinguishing under-represented roof types.

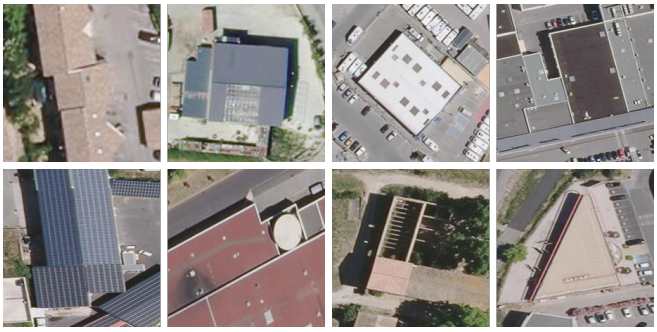

Fig. 1. Image examples of Aerial images used in our experiments

In Section 2 we propose an overview of different CL techniques and in Section 3 we present the results obtained by these techniques applied to aerial images for roof-type classification. Finally in Section 4, we discuss and confirm how the performances obtained by SimCLR are essentially caused by dataset bias.

The contributions of this work are:

1. An empirical observation of the strong impact of class imbalance and dataset bias on unsupervised CL pretraining strategies (especially SimCLR);
2. An empirical observation that the use of large minibatches during pretraining limits this problem, but is prohibitive in terms of computational resources;
3. An empirical analysis and comparison of the perfor-

This work was partially financed by IDEX Paris-Saclay IDI 2016, ANR-11-IDEX-0003-02, Office of Naval research grant N00014-17-1-2552, DGA Astrid project "filmer la Terre" nº ANR-17-ASTR-0013-01, MENRT. This work was also using HPC resources from GENCI–IDRIS (grant 2020-AD011011801) and from the "Mésocentre" computing center of CentraleSupélec and ENS Paris-Saclay supported by CNRS and Région Île-de-France (http://mesocentre.centralesupelec.fr/).

mances of recent CL methods in presence of imbalanced datasets, and a conclusion that nowadays BYOL [4] is the most suited for our application.

2. OVERVIEW OF CONTRASTIVE LEARNING FOR IMAGES REPRESENTATION TECHNIQUES

Despite all the work that has been done in developing the contrastive and predictive coding paradigm [5, 6, 7], fully supervised discriminative and generative techniques are still the mainly used ones today. The first successful use of contrastive learning is from van den Oord et al. [1].

Given an input sequence $X = \{x_1, \ldots, x_N\}$ of observations, the authors proposed an approach to learn a representation from it by predicting the future samples from a latent space representation computed by an auto-regressive model. Towards this aim, the authors seek to maximize the mutual information between the latent representation of the input signal and the future samples in the sequences by minimizing the CPC loss they introduced. They proved that using this technique, the network can learn usable representations for downstream tasks. Hénaff et al. [2] went a step further by adapting the paradigm to images. In their work, they adapted the loss by introducing negative samples that are taken from other locations in the image and other images from the mini-batch.

The second breakthrough was the introduction of SimCLR [3] as a simpler framework for CL, which proposed a new way of using the contrastive loss in order to encode the invariance to transformations in the learned representations. This means that two transformed versions of same images must have close representations. Thus, given N images $\{x_1, \ldots, x_N\}$, for each x_k we apply two different transformations to obtain two augmented versions of the same image x_{2k} and x_{2k-1}. The CPC loss function then writes

$$\mathcal{L}_{\text{CPC}} = \frac{1}{2N} \sum_{k=1}^{N} [\ell(2k-1, 2k) + \ell(2k, 2k-1)] \quad (1)$$

with

$$\ell(i, j) = -\log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_j)/\tau)}{\sum_{k \neq i} \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_k)/\tau)}, \quad (2)$$

where the variables $sim(\mathbf{z}_i, \mathbf{z}_j)$ are the cosine similarities between the representation vectors $\mathbf{z}_i := g_{encoder}(x_i)$ corresponding to different elements within the same batch. This idea was further developed in BYOL [4] , where the authors added another branch to the model, using rolling means of learned parameters at different training steps of the network. This new branch is used to encode one of the transformations of the original image and doesn't count in the gradient calculations for the loss as a stop gradient operator is used on the top of the branch. Lately, Caron et al. [8] presented SWAV, which is an improvement to this paradigm that simultaneously clusters the data while enforcing consistency between cluster assignments produced for different augmentations (or "views") of the same image, instead of comparing features directly as in contrastive learning.

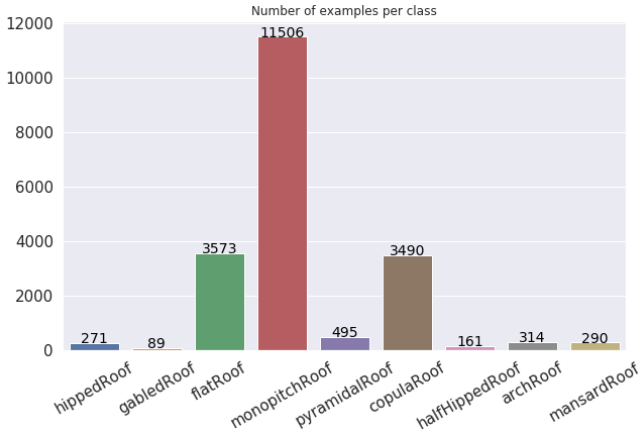

Fig. 2. Number of examples per class in the labeled dataset.

3. APPLICATION OF CL TECHNIQUES ON AERIAL IMAGES DATASET

3.1. Implementation details

We decided to test different contrastive learning methods and evaluate their performances using the test accuracy and F1-score on the downstream classification task. We tested SimCLR [3] and BYOL [4] and compared their results with the ones we obtained when training our network in a fully supervised manner (fine-tuning using a pretrained model). We did not include SWAV [8] in the comparison as its self-supervised pretraining was very difficult to stabilize, probably due to its sensibility to its many hyperparameters.

The experiments go as follows: we pre-train a randomly initialized network on a large unlabeled dataset of aerial images using SimCLR or BYOL. Then, we add a classification layer on the top of each obtained encoder network. We then train the new obtained models to classify the representations of the images. We use the classification accuracy as a score to compare these approaches with the baseline. On this particular dataset there was a choice to be made about whether or not freezing the weights learned in the first step before adding the classification layer. We decided not to freeze these weights for two main reasons:

1. It gives better results because we weren't able to match the batch sizes used in the reference papers. When a small batch size is used the learned representations are not linearly separable (see Fig 3). But they can be used as good initialization for fine-tuning.

2. The supervised baseline is also allowed to finetune the encoding network weights.

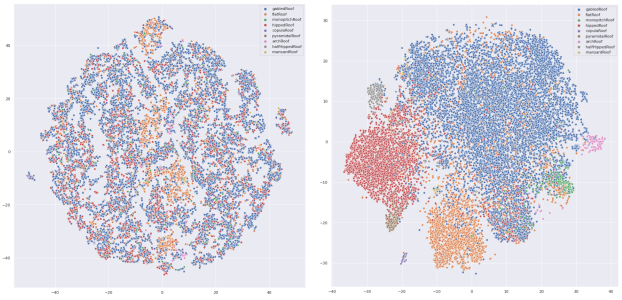

Fig. 3. T-SNE 2D projection of representations of the test set. Left: SimCLR (before fine-tuning), Right: Supervised.

	Accuracy/ F1-score
BYOL	**0.86/0.73**
SimCLR	0.79/0.51
Supervised	0.87/0.69

Table 1. Accuracy/F1-score for different CL algorithms compared to the fully supervised baseline.

The architecture used for all the unsupervised experiments is ResNet-34 [9]. We set the batch size to 256 and used the Adam optimizer [10]. We trained the models for 200 epochs for pretraining and 20 epochs for supervised fine-tuning.

3.2. Results

The results of this experiment are shown in Table 1. We can see that BYOL performs well and yields results that are the closest to the fully supervised baseline. In addition we observe that SimCLR performs poorly on this task. We believe that this is due to the fact that we used a small batch size (256) compared to the batch sizes used in [3]. In fact, When using large mini-batch sizes, mini-batches are more likely to contain a diverse set of images, which reduces the bias on each batch of images. Van den Oord et al. [1] proved that larger batch sizes result in larger mutual information between the input x_{t+k} and its representation c_t

$$I(x_{t+k}, c_t) \geq \log(N) - \mathcal{L}_N, \tag{3}$$

where N is the batch size, $\mathcal{L}_N$ is the CPC loss as defined in [1] and I is the mutual information. In contrast, when small mini-batch sizes are used the batches at each training step become very biased. Especially considering the initial large bias in the whole dataset. This translates into a lack of diversity in images in each batch which results in similar images being used as negative pairs, thus affecting the training process.

3.3. Sensibility to the used amount of labeled data

To analyze the robustness of these methods to the amount of labeled data used in the supervised fine-tuning step, we fine-tuned the self-supervised training models using varying ratios of annotated data. The results are shown in Table 1 (for 100% of labeled data) and Table 2. We can see that, despite the better results obtained with the fully supervised training using 100% of the labeled data, the models pretrained using BYOL perform better when using smaller amounts of labeled data and that the gap closes as we increase its ratio. This confirms the usefulness of using CL methods (and BYOL in particular in our case) for tasks that rely on remote sensing data (e.g Aerial Images) especially when we have a limited amount of annotated data.

	5%	10%	25%	50%	75%
BYOL	0.69/0.65	0.73/0.68	0.78/0.70	0.82/0.70	0.84/0.71
SimCLR	0.58/0.50	0.61/0.60	0.65/0.63	0.71/0.64	0.75/0.66
Supervised	0.60/0.62	0.64/0.64	0.71/0.66	0.79/0.67	0.85/0.68

Table 2. Classification accuracy/ F1-score obtained using different representation learning method varying the ratio of annotated data.

4. DISCUSSION: THE IMPORTANCE OF A BALANCED BATCH SAMPLING

In order to verify the claim that SimCLR actually performs badly due to the diversity problem, we did the following experiment: Using **only the labeled data**, we trained the SimCLR model two times, first using the full dataset, and second using the full dataset but introducing a batch sampler that, for each batch, ensures the class balance by oversampling less represented classes and undersampling the most represented ones. We then compared the learned representations using the linear classification test score **on the same test set** after transfer learning and fine-tuning. Results are reported in Table 3. In addition, we plotted the confusion matrix of the classification model when the SimCLR step was applied with and without batch sampler (see Fig 4).

We observed that when we do not use the batch sampler in pretraining, the classification model obtained after fine-tuning is heavily affected by the class imbalance. It predicts mainly the most represented classes. We also observed that using the same technique for the classification step badly affects the test accuracy. This is due to the fact that the most represented class in the dataset is heavily penalized by batch sampling in the supervised training procedure. From these results it seems that controlling the class balance for the Contrastive Learning step hugely affects the classification accuracy on learned representations even with relatively small batch size.

To further confirm the effect of the batch sampling, we realized the same experiments on CIFAR-10 dataset and on a long-tailed version of the same dataset. This time using two different batch sizes (200 and 800). The results are shown in Table 4.

Here we see that, even if we do not attain the performance

	For SimCLR	For classification	Test accuracy
Unsupervised	-	-	0.77
	✓	✓	0.67± 0.04
	✓	-	0.85± 0.03
Supervised	N/A	-	**0.87**

Table 3. Linear classification accuracy comparison. The second and third columns represent whether we used the batch sampling for each training step.

Dataset	Batch size	Supervised	SimCLR without batch sampler	SimCLR with batch sampler
LT CIFAR-10	200	**0.91**	0.82	0.86
CIFAR-10	200	**0.90**	0.86	0.88
LT CIFAR-10	800	**0.89**	0.86	0.88
CIFAR-10	800	**0.86**	0.86	0.85

Table 4. Classification accuracies after SimCLR pretraining on CIFAR-10 and Long-Tailed (LT) CIFAR-10 (factor = 0.5) with and without batch sampler.

Fig. 4. Confusion matrices, from left: classification after SimCLR with batch sampler, right: classification after SimCLR without batch sampler.

of the fully supervised model, the batch sampling allows us to largely improve the accuracy on the long-tailed dataset. Especially with the smaller batch size, as the improvement is less meaningful when using 800 as batch size. The improvement is also less meaningful on the default CIFAR-10 dataset. We believe that this result confirms our intuition about the effect of class imbalance on the SimCLR training and opens a promising research direction to find better ways to assure more diversity in the batches. Possibly without the need of the labels like in our experiment with batch sampling.

5. CONCLUSION AND FUTURE WORK

In this work we compared different CL techniques on an aerial image dataset for roof-type classification. The characteristic of this type of dataset being large bias and class imbalance, we have empirically proven that these properties can have a negative impact on certain contrastive learning techniques (SimCLR in our case), Particularly when we use small batch sizes, Although more advanced algorithms like BYOL successfully overcome this type of issues. We believe that these results open a promising research direction on how to improve the reliability of CL algorithms on biased datasets in order to attain the performances of fully supervised trained models.

6. REFERENCES

[1] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[2] O. J. Hénaff, A. Srinivas, J. De Fauw, A. Razavi, C. Doersch, SM Eslami, and A. van den Oord, "Data-efficient image recognition with contrastive predictive coding," *arXiv preprint arXiv:1905.09272*, 2019.

[3] T. Chen, S. Kornblith, Med. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," feb 2020.

[4] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, et al., "Bootstrap your own latent: A new approach to self-supervised learning," *arXiv preprint arXiv:2006.07733*, 2020.

[5] B. Atal and M. Schroeder, "Adaptive predictive coding of speech signals," *Bell System Technical Journal*, vol. 49, no. 8, pp. 1973–1986, 1970.

[6] P. Elias, "Predictive coding–i," *IRE Transactions on Information Theory*, vol. 1, no. 1, pp. 16–24, 1955.

[7] R. Rao and D. Ballard, "Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects," *Nature neuroscience*, vol. 2, no. 1, pp. 79–87, 1999.

[8] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *arXiv preprint arXiv:2006.09882*, 2020.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015.

[10] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.