Self-supervised Contrastive Learning for Volcanic Unrest Detection

Bountos Nikolaos Ioannis[12], Ioannis Papoutsis, *Member, IEEE*[1], Dimitrios Michail[2], and Nantheera Anantrasirichai[3]

[1]Institute of Astronomy, Astrophysics, Space Applications & Remote Sensing, National Observatory of Athens
[2]Department of Informatics & Telematics, Harokopio University of Athens
[3]Department of Computer Science, University of Bristol

Abstract

Ground deformation measured from Interferometric Synthetic Aperture Radar (InSAR) data is considered a sign of volcanic unrest, statistically linked to a volcanic eruption. Recent studies have shown the potential of using Sentinel-1 InSAR data and supervised deep learning (DL) methods for the detection of volcanic deformation signals, towards global volcanic hazard mitigation. However, detection accuracy is compromised from the lack of labelled data and class imbalance. To overcome this, synthetic data are typically used for finetuning DL models pre-trained on the ImageNet dataset. This approach suffers from poor generalisation on real InSAR data. This letter proposes the use of self-supervised contrastive learning to learn quality visual representations hidden in unlabeled InSAR data. Our approach, based on the SimCLR framework, provides a solution that does not require a specialized architecture nor a large labelled or synthetic dataset. We show that our self-supervised pipeline achieves higher accuracy with respect to the state-of-the-art methods, and shows excellent generalisation even for out-of-distribution test data. Finally, we showcase the effectiveness of our approach for detecting the unrest episodes preceding the recent Icelandic Fagradalsfjall volcanic eruption.

Index Terms

deep learning, self-supervised, contrastive learning, SimCLR, volcano, InSAR, Fagradalsfjall eruption

I. Introduction

Globally, 800 million people live within 100 km of a volcano [1]. Improvements in forecasting volcanic activity have been shown to reduce fatalities due to volcanic eruptions [2]. Hence, several volcano observatories are set-up globally, including the Geohazard Supersites and Natural Laboratories initiative. However, a significant proportion of the $\sim$1,500 holocene volcanoes has no ground-based monitoring, although the deformation at volcanoes is statistically linked to eruption [3], which can be detected prior to the event [4].

Interferometric Synthetic Aperture Radar (InSAR) data from the Sentinel-1, 6-day repeat-cycle, satellite allows the systematic monitoring of volcanic unrest at a global scale. Such abundance of InSAR data have the potential to enable observatories to monitor volcanic activity without additional costs. Fringes detected in wrapped interferograms over volcanoes indicate the onset of deformation, usually due to magma chamber fill-in at depth. Associating fringes with deformation is non-trivial; atmospheric signals may also give rise to such fringes, which may lead to false positive identifications. Their effect is also amplified in the presence of strong topography, which is usually the case for volcanic domes.

Recent studies have proposed the use of supervised Deep Learning (DL) architectures to automatically detect the presence of ground deformation triggered by volcanic unrest, within single interferograms. Anantrasirichai et al. [5] were the first ones to use DL on short-term wrapped interferograms to detect rapid deformation. They rely on heavy data augmentation and employ a transfer learning strategy using AlexNet Convolutional Neural Network (CNN) architecture, pre-trained on ImageNet [6]. Synthetically generated interferograms, based on analytic

This work has received funding from the European Union's Horizon2020 research and innovation project DeepCube, under grant agreement number 101004188.

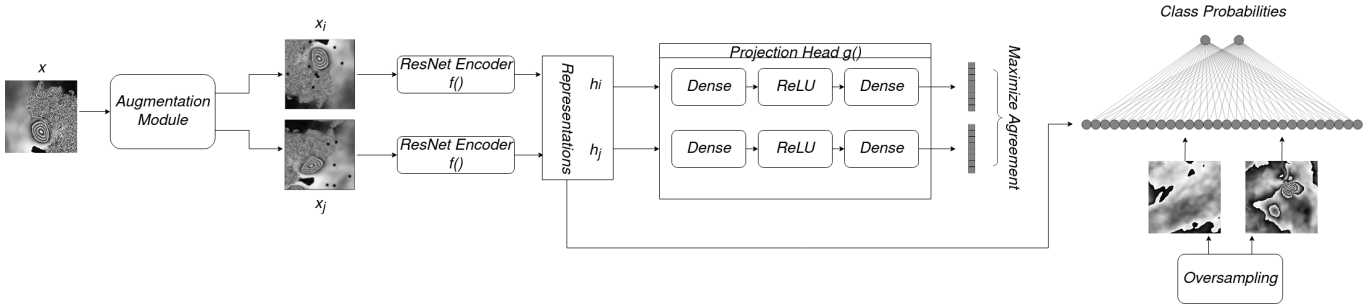

Fig. 1: The proposed pipeline. First, we use unlabeled InSAR data to learn feature representations with the SimCLR [13] self-supervised framework and then attach a linear classifier for the supervised training with a few labels.

forward models of magma chambers (e.g. Mogi, dykes, sills), have also been used to train the same network [7], reaching 86% F1-score on real data. Valade et al. [8] train a custom CNN architecture on synthetic wrapped interferograms, but test it to only a few real interferograms. A CNN workflow has also been proposed to identify surface deformation associated with an earthquake [9], using synthetic wrapped and unwrapped interferograms. They report an accuracy for their best model of 85%, tested on 32 real InSAR interferograms, yet in principle earthquake induced interferometric signals are denser and clearer than the volcanic ones. Finally, volcanic deformation detection from a single interferogram using a combination of synthetic and augmented real InSAR data has been performed by Gaddes et al. [10]. They apply a VGG-based model, while they split the target class "deformation", to two sub-classes: Sill/Point and Dyke and report an overall accuracy of 83%.

A common issue encountered by all studies is the scarcity of training data for the positive class, i.e. interferograms with volcanic deformation. Hence, most works cope with class imbalance via heavy data augmentation engineering and the creation of positive class synthetic data for supervised learning. Supervised learning though requires big curated datasets to work well. To counter this, researchers have used classical architectures pretrained on large unrelated datasets. We argue that transfer learning from computer vision tasks provide less meaningful information for the task of volcanic activity detection when compared to features learnt from related data. Indeed the studies discussed above, report that they struggle to generalise well for real, unseen InSAR data.

Recently, the AI community shifted its focus to resolve these shortcomings of supervised learning, towards unsupervised/self-supervised training schemes. The goal is to exploit the information hidden inside the data and produce features without any human supervision that can generalize well for different classification tasks. In remote sensing some recent works highlight the value of these approaches [11][12].

In this letter, we propose a self-supervised, contrastive learning framework based on SimCLR [13] to solve the deformation/non-deformation binary classification problem, using an unbalanced, real, wrapped InSAR dataset. We avoid using unwrapped interferograms or time-series data in order to have faster classification response, which is particularly useful in volcano observatories for near real-time monitoring. Our approach exploits the abundance of unlabelled InSAR data to learn quality visual features, which can be used by a simple linear supervised classifier for the detection task.

The contributions of our work are as follows:

- We are the first to introduce a self-supervised learning framework for volcanic activity detection.
- We propose a training pipeline that does not rely on the generation of massive augmented, synthetic or manually annotated InSAR data.
- We demonstrate that training on a large unlabeled InSAR dataset in a self-supervised manner provides more quality features than using pre-trained models from ImageNet.
- Experimental results show that models trained with this framework have the ability to generalize better, even for InSAR data drawn from a different distribution with respect to the training set.
- We provide the first generic feature learning model for InSAR, which can be used for different downstream tasks.

II. Contrastive self-supervised learning pipeline

Our approach is set-up as an instance discrimination task where every image in the dataset belongs to its own class. Our pipeline is a two step training process. First, it consists of an encoder trained in a self-supervised manner, and second of a fully connected layer attached on top of the encoder for the supervised classification task. For the self-supervised training we utilize the recently introduced SimCLR [13] framework. SimCLR learns representations by trying to maximize the similarity of two augmented views of the same example in the latent space. The main components of the adopted framework are the following:

- A stochastic data augmentation module that creates two different transformations of the same input wrapped interferogram patch. For every patch x the augmentation module creates two augmented views $\tilde{x}_i, \tilde{x}_j$. In our experiments, we use Horizontal and Vertical Flips, Cutout, Multiplicative Noise, Elastic Transformation, Gaussian Blur and Gaussian Noise. Each view is generated from a random combination of these augmentations.
- An encoder $f(\cdot)$ for the representation extraction from the augmented patches. There is no constraint on the choice of the encoder. Following [13], we use the ResNet [14] architecture. The representation is then extracted from the output after the average pooling layer, $h_i = ResNet(\tilde{x}_i)$.
- A projection head $g(\cdot)$, that maps the encoder's representation to the space where the contrastive loss is calculated. We, like [13] use a multilayer perceptron (MLP) with one hidden layer and a ReLU activation function. Thus, $z_i = g(h_i) = W^{(2)}\sigma(W^{(1)}h_i)$, where W^1 and W^2 represent the weight matrices and σ is the ReLU function.
- The contrastive loss estimation. Given a set $\{\tilde{x}_k\}$ which contains, among others, a positive pair of augmented interferogram patches $\tilde{x}_i$ and $\tilde{x}_j$, the contrastive loss aims to guide our algorithm to identify $\tilde{x}_j$ in $\{\tilde{x}_k\}_{k\neq i}$ for a given $\tilde{x}_i$.

The training pipeline can be seen in Figure 1 and is as follows. From a batch of size N we create 2N samples using the augmentation process defined above. The augmented InSAR patches created from the same original patch serve as a positive example and the rest 2(N-1) patches in the batch constitute the negative examples. The process can be summarized with the following graph: $x_k \xrightarrow{t\in T} \tilde{x}^k_{i,j} \xrightarrow{f(\cdot)} h^k_{i,j} \xrightarrow{g(\cdot)} z^k_{i,j}$, where T is the set of augmentations. We define the similarity function $sim(x, y)$ as the cosine similarity between x and y. For each minibatch we attempt to minimize the following contrastive loss function:

$$l_{i,j} = -log\frac{e^{sim(z_i,z_j)/\tau}}{\sum_{k=1}^{2N} \mathbb{1}_{[k\neq i]} e^{sim(z_i,z_k)/\tau}}, \tag{1}$$

where $\mathbb{1}_{[k\neq i]} \in \{0, 1\}$ is 1 if $k \neq i$ and 0 otherwise and τ is a temperature parameter. τ scales the input and expands the range of values of the cosine similarity. We set $\tau = 0.5$ and the final loss is calculated on all positive pairs in a mini batch.

Finally, using the encoder that was trained in a self-supervised manner we proceed with the classification task, by freezing the parameters of the encoder and attach a trainable linear classifier on top. We propose an oversampling approach for the supervised classification task, where we randomly choose a class from which the next sample will be drawn. For a batch size N, this process takes place N times. In this approach one sample might be seen more than once in each epoch, thus preventing the domination of the larger class.

III. Experiments

All data and code presented in this section are published at the project's repository: (*https://github.com/ngbountos/DeepCubeV*

A. Datasets

We use only real data that come from two different sources, which we symbolize with S1 and C1 (Table I). S1 dataset was provided by the authors of [5] and [7], and contains Sentinel-1 wrapped InSAR patches from 16 volcanoes globally. The S1 dataset is highly imbalanced, containing a large number of negative examples but very few positive ones ($\sim$2%). C1 was manually collected by us from the LiCSAR online InSAR repository [15] over 5 volcanoes: Taal, Cerro Azul, Fagradalsfjall, Etna and Ale Bagu, and is much more balanced. We use only the S1 dataset for training our models, while we create two different test datasets. The first one contains 64 balanced samples drawn from S1, and the entire C1 dataset serves as a, second, evaluation dataset.

TABLE I: Overview of the datasets used in this work. For the self-supervised task, the positives and negatives of S1 are employed together without labels. C1 is used only for evaluation.

Data Source	Train		Test		Total
-	Positive	Negative	Positive	Negative	
S1	150	7386	32	32	7600
C1	-	-	404	365	769

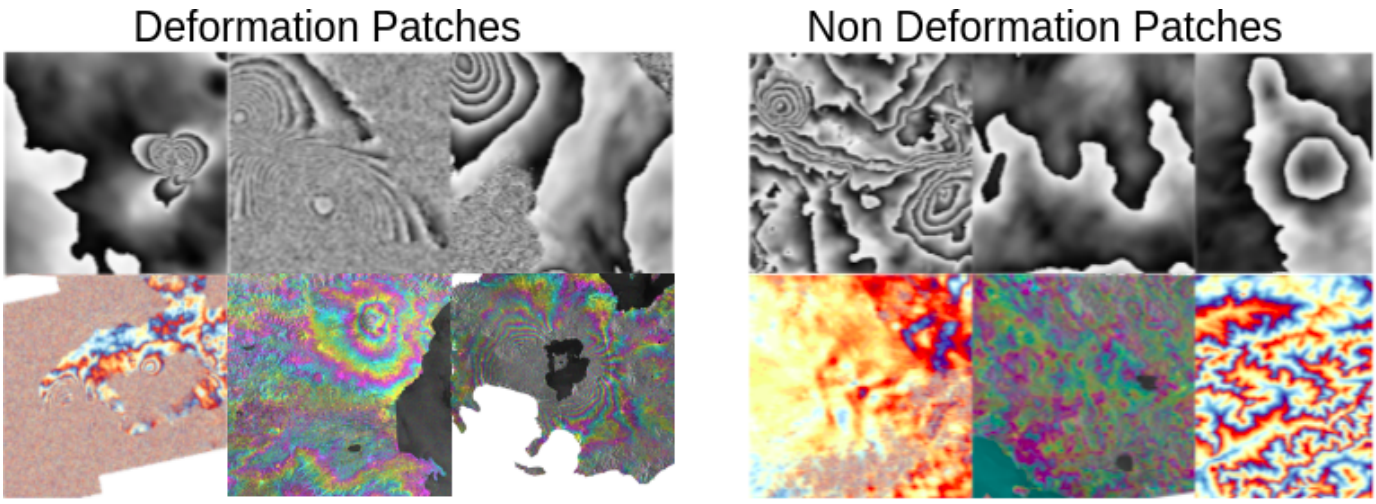

Fig. 2: Sample interferogram patches from both data sources. The first and second rows contain samples from S1 and C1 sources respectively. C1 dataset is diverse - from left-to-right and for the deformation class: 1) Cerro Azul unfiltered, interferometric phase only, 2) Etna descending interferogram, Goldstein filtered, phased and amplitude overlay, and 3) Taal, ascending interferogram, water masked interferometric phase, Goldstein filtered.

The two test sets S1 and C1 are quite different. Figure 2 shows examples of deformation and non-deformation samples from both sources. C1 test dataset is highly diverse as we have included 1) wrapped interferograms from both descending and ascending viewing geometries, 2) unfiltered and Goldstein phase-filtered interferograms, and 3) pure phase-only interferograms and interferograms where the phase is overlaid with SAR amplitude. S1 dataset is much more harmonised and therefore S1 and C1 are considered to be drawn from distributions with different characteristics with respect to noise level and visual features. Such challenging test samples are used to evaluate the generalization performance of our self-supervised approach.

B. Contrastive learning model performance

We evaluate the performance of the encoder trained with our proposed self-supervised approach (Section II) and compare it to the pretrained features from ImageNet [6]. We experiment with different scales of the ResNet architecture and utilize the linear evaluation protocol, i.e. we freeze the encoder parameters and fine-tune a simple linear classifier on top of the network. The linear evaluation protocol is a standard way to assess the quality of the learnt representations [16]. To speedup convergence for the self-supervised stage, we initialize the weights of the encoder with the parameters learnt from ImageNet. We then retrain all layers with the self-supervised method for 200 epochs using an unlabelled version of the S1 dataset. At the fine-tuning step we apply the oversampling approach (Section II) to the labeled S1 dataset. No oversampling is performed for the self-supervised stage, as it assumes no class knowledge.

Table II shows how ResNet architectures with different capacities compare in the two different test setups. The models we examine are ResNet18, ResNet34 and ResNet50. It is noteworthy that the results using the features learnt from the contrastive learning framework were obtained after only 1 to 3 epochs of fine-tuning the linear classifier. We found that setting the learning rate between 0.001 and 0.005 works best. On the contrary, the networks that used the pre-trained encoder from ImageNet required 75 epochs to converge. We set the learning rate to 0.001. For

the fine-tuning step we use a batch size of 112. At the pre-training stage, we use the largest possible batch size depending on the architecture. We set the batch size to 32 for ResNet50 and 112 for ResNet18 and ResNet34.

We conduct two additional experiments using ResNet50, our best performing backbone encoder. First, we train SimCLR from scratch using random initialization for 300 epochs to show that the performance gain comes from self-supervised training on InSAR data alone. We publish this model on the project's repository. Second, we test MoCo [16], a different self-supervised approach to show that performance gain can be achieved from different self-supervised methods as well.

TABLE II: Experiment results on both test sets. ACC, FP, TP, FN, TN, F1, P and R, stand for overall accuracy, false positives, true positives, false negatives, true negatives, f1-score, Precision and Recall, respectively.

	S1								C1							
Model	ACC	FP	TP	FN	TN	F1	P	R	ACC	FP	TP	FN	TN	F1	P	R
ResNet18-ImageNet	81%	12	32	0	20	0.841	0.727	1	64%	0	132	272	365	0.491	1	0.326
ResNet18-SimCLR	**84%**	9	31	1	23	0.860	0.775	0.968	70%	0	178	226	365	0.611	1	0.440
ResNet34-ImageNet	82%	10	31	1	22	0.848	0.756	0.968	70%	3	181	223	362	0.615	0.983	0.448
ResNet34-SimCLR	82%	11	32	0	21	0.853	0.744	1	**91%**	4	339	65	361	0,907	0.988	0.839
ResNet50-ImageNet	82%	10	31	1	22	0.848	0.756	0.968	63%	1	125	279	364	0.471	0.992	0.309
ResNet50-SimCLR	85%	8	31	1	24	0.872	0.794	0.968	**91%**	10	347	57	355	**0.911**	0.971	0.858
ResNet50-SimCLR-Scratch	85%	8	31	1	24	0.872	0.794	0.968	86%	2	306	98	363	0.859	0.993	0.757
ResNet50-Moco	79%	13	32	0	19	0,831	0.711	1	82%	0	267	137	365	0.795	1	0.660
AlexNet	82%	9	30	2	23	0.844	0.769	0.937	72%	49	244	160	316	0.699	0.832	0.603
VGG16	**85%**	6	29	3	26	0.865	0.828	0.906	64%	39	171	233	326	0.556	0.814	0.423
ViT-ImageNet	90%	3	29	3	29	0.906	0.906	0.906	59%	5	96	308	360	0.343	0.950	0.343
DenseNet121-ImageNet	82%	9	30	2	23	0.844	0.769	0.937	54%	0	53	351	365	0.231	1	0.131
InceptionV4-ImageNet	**92%**	4	31	1	28	**0.924**	0.885	0.968	69%	23	196	208	342	0.628	0.894	0.485

Additionally, we compare our models with the state-of-the-art - all published methods use supervised approaches. We train AlexNet's and VGG16's final layer for 50 epochs, while keeping the rest of the layers freezed, with the ImageNet pre-trained weights. We also examine three more methods, popular in computer vision i.e Vision Transformer (ViT), DenseNet121 and Inception-V4 (Table II). We use oversampling and random rotation augmentation for all methods.

C. Discussion

The results summarised at Table II show that the models trained with the SimCLR method performed better or comparable with the respective ones pre-trained with ImageNet, for each test dataset and for each ResNet encoder architecture. This is significant; it highlights the fact that training in a self-supervised approach with 7,536 unlabeled samples only (Table I), drawn from a distribution of wrapped interferograms, provides better quality features than using models pre-trained in a supervised way from $\sim$1.5 million ImageNet images. It is even more impressive that our proposed self-supervised learning technique produced models that required only 1-3 epochs of finetuning to achieve these results, versus the 75 epochs needed for the Imagenet pre-trained model, for the same task.

The major enhancement in our approach however, shows itself at the C1 dataset. The high quality of the learnt InSAR data representations is clearer in the second half of Table II that summarises the experiments on the C1 test dataset, which is drawn from a distribution with different characteristics with respect to the training set (Section III-A). While for S1 dataset the best ImageNet and SimCLR models provide 92% and 85% overall accuracy respectively, for C1 dataset the corresponding accuracies are 70% and 91%. The supervised ImageNet model struggles to resolve the required features from the new InSAR data distribution and overall, this large performance gap underlines the ability of the self-supervised model to generalize better. To further validate this we construct 731 synthetic interferograms using SyInterferoPy [10] over a collection of subaerial volcanoes. Deformation patterns include those due to dykes, sills, and Mogi sources. Our ResNet50-SimCLR model reaches 88.9%, while ResNet50-ImageNet 69.9% and InceptionV4-ImageNet 66.2% true positive rate respectively. Again the self-supervised learnt features generalize better.

The last five rows of Table II provide a comparison between ResNet50-SimCLR, our best performing encoder, and architectures on the same task, proposed by the state-of-the-art studies of Section I and other popular methods

used in computer vision. Again, the proposed method generalizes better, as seen in the C1 part of the table. On S1 all models perform well with Inception-V4 achieving the best accuracy. On C1, our method reaches 91% overall accuracy and 0.911 F1 score based on the linear evaluation protocol.

In addition, our pipeline did not make use of massive augmented, or synthetically generated datasets, as opposed to the state-of-the-art approaches (Section I). Equally important, the achieved performance has been reached with a training set containing just 150 manually annotated InSAR patches with deformations. The potential from exploiting large unlabelled InSAR datasets in a self-supervised approach is at least promising. Given the recent availability of online repositories, such as LiCSAR used in this work, that contain and produce hundreds of interferograms over volcanoes globally, the deployment of this pipeline on a large distributed system, increasing the effective batch size is a natural next step. In fact, greater computational resources lead to better models. Using larger batch sizes, training longer and optimizing the stochastic augmentations used for SimCLR, improves the performance of contrastive learning [13]. Furthermore, our analysis showed that Elastic Transformation is especially important for data augmentation, potentially due to the nature of fringe patterns within InSAR data, either these patterns are related to deformation, atmospheric disturbances, topographical errors, or orbital ramps, etc. In our work we also use the maximum batch size possible depending on the model's architecture and available memory.

IV. Fagradalsfjall volcanic eruption case study

We apply our approach on Fagradalsfjall volcano at Reykjanes Peninsula, Iceland. We focus on two recent unrest episodes. Triggered by dyke intrusions, the inflation episodes took place in mid-January 2020, and early March 2021. The latter episode led to an effusive eruption, on 19 March 2021, – the first known eruption on the peninsula in about 800 years.

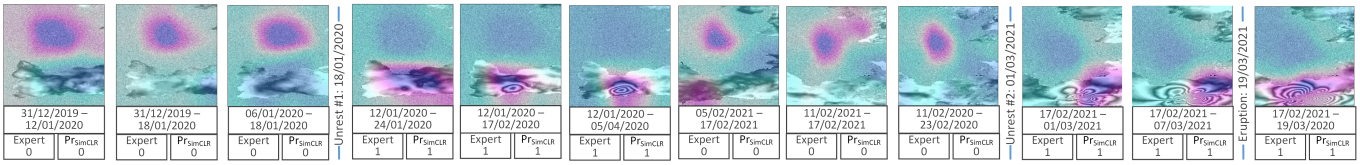

Fig. 3: Visualization of ResNet50 activations on Fagradalsfjall volcano. Pink represents the area that affected the network's decision the most. Both unrest episodes are shown in chronological order. Pr_{SimCLR} and Expert are the predictions made by our method and the InSAR expert, respectively (1=positive, 0=negative).

In order to simulate a real, working product, we merge all our datasets and collect some extra InSAR patches to finetune our models and produce a quality classifier. We unfreeze the fully connected layer as well as layer 3 and 4 of the ResNet50 SimCLR encoder. We train for 2 epochs and reduce the learning rate to 0.0005. In total the new training set contains 614 deformation and 7872 non deformation patches. We feed the network with two time-series of wrapped interferograms, one for each unrest event.

Figure 3 presents the model classification decision for each single interferogram vis-à-vis the decision from an InSAR expert. In addition, the figure shows the areas of the patch that affect the most the decisions of the final, fully connected layer. This visualization was produced using the Class Activation Mapping (CAM) technique [17]. There are two key observations drawn from this use case. First, the model trained with the proposed method focuses on the correct patterns (the fringes) of the interferogram patch. Second, the pipeline correctly captures the start of both unrest episodes, triggering a potential alarm.

V. Conclusion

In this work, we implemented a pipeline to train binary classification models for volcanic unrest detection utilizing unlabelled InSAR datasets, thus without the the need to create huge labelled datasets or generate error-prone synthetic data. We provided proof for the superiority of the self-supervised learnt features when compared to models pre-trained from ImageNet, and the ability of our approach to generalise effectively even for out-of-distribution test samples. Our approach outperforms state-of-the-art supervised methods.

Volcanic unrest early warning is of major importance for civil protection authorities and volcano observatories. Setting-up alert mechanisms enhances response effectiveness and allows for scientists to deploy critical in-situ

monitoring equipment to assess more accurately volcanic hazard. We highlighted that this could be implemented in the case of the 2020-2021 Fagradalsfjall volcano unrest and eruption.

Finally, we believe that there is much potential in our self-supervised approach, given the abundance of InSAR data produced regularly by the Sentinel-1 missions. Exploiting the information they contain in a self-supervised way, while labelling only a small subset paves the way towards a global volcanic unrest detection system, but may also be applicable to a plethora of other remote sensing applications and tasks.

REFERENCES

[1] *Global Volcanic Hazards and Risk*. Cambridge University Press, 2015.

[2] M. R. Auker, R. S. J. Sparks, L. Siebert, H. S. Crosweller, and J. Ewert, "A statistical analysis of the global historical volcanic fatalities record," *Journal of Applied Volcanology*, vol. 2, p. 2, Feb. 2013.

[3] J. Biggs, S. Ebmeier, W. Aspinall, Z. Lu, M. Pritchard, R. Sparks, and T. Mather, "Global link between deformation and volcanic eruption quantified by satellite imagery," *Nature communications*, vol. 5, no. 1, pp. 1–7, 2014.

[4] M. A. Furtney, M. E. Pritchard, J. Biggs, S. A. Carn, S. K. Ebmeier, J. A. Jay, B. T. McCormick Kilbride, and K. A. Reath, "Synthesizing multi-sensor, multi-satellite, multi-decadal datasets for global volcano monitoring," *Journal of Volcanology and Geothermal Research*, vol. 365, pp. 38–56, 2018.

[5] N. Anantrasirichai, J. Biggs, F. Albino, P. Hill, and D. Bull, "Application of machine learning to classification of volcanic deformation in routinely generated insar data," *Journal of Geophysical Research: Solid Earth*, vol. 123, no. 8, pp. 6592–6606, 2018.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[7] N. Anantrasirichai, J. Biggs, F. Albino, and D. Bull, "A deep learning approach to detecting volcano deformation from satellite imagery using synthetic datasets," *Remote Sensing of Environment*, vol. 230, p. 111179, 2019.

[8] S. Valade, A. Ley, F. Massimetti, O. D'Hondt, M. Laiolo, D. Coppola, D. Loibl, O. Hellwich, and T. R. Walter, "Towards global volcano monitoring using multisensor sentinel missions and artificial intelligence: The mounts monitoring system," *Remote Sensing*, vol. 11, no. 13, p. 1528, 2019.

[9] C. M. Brengman and W. D. Barnhart, "Identification of surface deformation in insar using machine learning," *Geochemistry, Geophysics, Geosystems*, p. e2020GC009204, 2021.

[10] M. Gaddes, A. Hooper, and F. Albino, "Simultaneous classification and location of deformation in sar interferograms using deep learning," 2021.

[11] H. Jung, Y. Oh, S. Jeong, C. Lee, and T. Jeon, "Contrastive self-supervised learning with smoothed representation for remote sensing," *IEEE Geoscience and Remote Sensing Letters*, pp. 1–5, 2021.

[12] D. Guo, Y. Xia, and X. Luo, "Self-supervised gans with similarity loss for remote sensing image scene classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 2508–2521, 2021.

[13] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, pp. 1597–1607, PMLR, 2020.

[14] 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*, pp. 770–778, 2016.

[15] M. Lazecký, K. Spaans, P. J. González, Y. Maghsoudi, Y. Morishita, F. Albino, J. Elliott, N. Greenall, E. Hatton, A. Hooper, D. Juncu, A. McDougall, R. J. Walters, C. S. Watson, J. R. Weiss, and T. J. Wright, "Licsar: An automatic insar tool for measuring and monitoring tectonic and volcanic activity," *Remote Sensing*, vol. 12, no. 15, 2020.

[16] 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*, pp. 9729–9738, 2020.

[17] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2921–2929, 2016.