AVALANCHE DETECTION IN SENTINEL-1 RADAR IMAGES USING CONVOLUTIONAL NEURAL NETWORKS

Per Egil Kummervold[1,*], Eirik Malnes[1], Markus Eckerstorfer[1], Ingar M. Arntzen[1], Fillipo Bianchi[1]

[1]*Norut Northern Research Institute, Tromsø, Norway*

ABSTRACT: Knowledge about frequency and location of avalanche activity is important for avalanche forecasting and hazard mapping. Traditional field monitoring has limitations especially when surveying large remote areas. Thus avalanche detection in Sentinel-1 radar satellite imagery has been developed in recent years as an alternative. Current state-of-the-art automatic signal processing results in an accuracy of roughly 80%, but has in problematic cases (snow turning from wet to dry) an accuracy below 50% when compared to manual interpretation. We thus explored the use of convolutional neural networks (VGG-19 and AConvNets) in detecting avalanches in radar images, and evaluated if these networks were able to outperform currently used radar image classification. The CNN's produced consistently accuracies above 90%. While conventional signal processing seems to fail on images that are easily categorised by human experts, the neural networks seem to have problems with the same images that are also considered borderline cases by a human expert. It is likely that enlarged and improved datasets, as well as transferred learning, can increase the accuracy even more.

Keywords: Sentinel-1, avalanche detection, convolutional neural network, artificial intelligence

1. INTRODUCTION

1.1. Current state-of-the-art avalanche detection

Knowledge about avalanche activity in space and time is important for avalanche forecasting and hazard mapping. Traditional field monitoring of avalanche activity, however, has limitations due to often reduced visibility and difficulties in frequently monitoring inaccessible areas. This can lead to high uncertainty in determining avalanche risk.

As an alternative, synthetic aperture radar (SAR) data from Sentinel-1 has been used for avalanche detection in recent years. The use of SAR data allows to monitor large areas, unaffected by clouds and light conditions. Temporal change detection combined with edge detection are used to identify avalanches in radar images (Vickers et al., 2017). This state-of-the art avalanche detection method is currently used in operational avalanche detection in Norway.

Avalanches are detectable due to their relatively higher backscatter than the surrounding snowpack (edge detection), and their increased backscatter when comparing images with identical SAR geometry six days apart (temporal change detection). The high relative backscatter stems from the debris' high surface roughness compared to undisturbed snow (Eckerstorfer and Malnes, 2015).

*Corresponding author address:
Per Egil Kummervold, Norut Northern Research Institute, P.O. Box 6434, Tromsø Science Park, 9294 Tromsø, Norway;
email: per.egil.kummervold@norut.no

While an experienced operator is able to identify avalanches in SAR change detection composites with high confidence, automatic signal processing produces an average accuracy of roughly 80% (Vickers et al., 2017). Accuracies of over 90% are achieved in some ideal cases (snow turning from dry to wet), but in more problematic cases (snow turning from wet to dry), accuracy's drop below 50%. The large span in accuracy are due to a very dynamical radar signal from variable snow conditions (wet/dry snow) effecting the temporal change detection method. Another major issue is that while an operator can recognize an avalanche simply based on its shape, this is difficult to incorporate into traditional signal processing algorithms, as no avalanche is alike.

We believe that the detection method developed by Vickers et al. (2017) is close to reaching its 'hard limit' in terms of average achievable accuracy. We therefore hypothesize that by using convolutional neural networks for avalanche detection in SAR images, higher average accuracies can be achieved.

1.2. Basic concept of convolutional neural networks

Neural networks, also referred to as artificial neural networks, are a form of machine learning patterned after how neurons operate in the human brain. The processing nodes are the individual neurons ordered in multiple layers. Each neuron is assigned a weight used for calculating whether the individual neuron should fire. The network is then fed data input (in our case SAR data) to see how well it can

predict the output labels (in our case 'avalanche'/'no avalanche'). This process is repeated multiple times and based on the results, a gradient is calculated to optimise the neurons weights (backpropagation).

Convolutional neural network (CNN) is a special type of neural network that excels in extracting abstract features from raw images by applying sequentially convolutional and pooling layers. Convolutional layers contain spatial filters which are adapted to the data to solve a target task, while pooling layers, shrink the spatial features map to generate a more abstract representation (see Figure 2). CNNs are capable of capturing, at the same time, the non-linear relationships between the image channels and spatial patterns arising from the neighborhood. Their successful application in various remote sensing tasks has grown considerably in the last years (Zhu et al., 2017)

2. DATA AND METHODS

2.1. Dataset and preparatory work

The test dataset consists of eight Sentinel-1 images (activity images) with corresponding images of similar geometry and orbit from 6 days earlier (reference images). From these eight images, we constructed four data channels using the polarization of the radar images, where vv_1 and vv_2 stand for VV polarization in the activity image and reference image respectively and vh_1 and vh_2 for VH polarization. The Sentinel-1 images had a typical dimension of around 10.000 x 5.000 pixels, where each pixel measures 20 x 20 meter. In order to reduce the number of false alarms we masked out terrain where avalanches cannot be detected or occur (e.g. radar shadow and layover areas, dense forest, water bodies, too steep or too far away from a slope).

We trained neural networks to categorize small image slices into the labels 'avalanche' and 'non-avalanche'. Training used pre-labelled (manual interpretation) from an expert (see example in Figure 1). Each original SAR-image was split into slices of 64 x 64 pixels, making each slice 1280 x 1280 meters. The relative small size of the slices was chosen to reduce the number of avalanches per slice, to have a larger dataset of images and to allow using larger batch sizes, while at the same time keeping the memory consumption relatively low. However, a major drawback of using smaller slices was that they likely contained only a portion of the avalanche and that some spatial information useful for the detection was potentially lost.

An expert conducted manual identification in the eight Sentinel-1 change detection pairs and delineated avalanche debris. To facilitate interpretation, RGB images were constructed where the two VV polarization images vv_1 and vv_2 were put into the RGB channels as [vv_2, vv_1, vv_2]. Avalanches would then appear as green features in the rendered image (Figure 1). Slices that contained more than

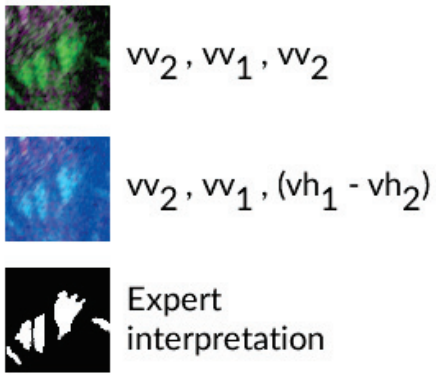

Figure 1: Examples of RGB images with polarization information in different channels and associated expert interpretation

10 pixels and marked as 'avalanche' by the operator, were included in the training set for avalanches. For each image a corresponding number of 'no avalanche' images were randomly selected. In both cases slices with less than 50% valid data were excluded from the training dataset.

After data preparation, the total dataset consisted of 817 slices with avalanches and 817 without avalanches. The training dataset consisted of 80% of these images (N=1308), while the remaining 326 images were used only for testing of the trained network and quantifying its accuracy.

2.2. Convolutional neural networks

Two different convolutional neural networks were evaluated (Figure 2). The first model is called VGG-19 (Simonyan and Zisserman, 2014). It is based on an up to 19-layer deep CNN that is commonly used for categorising natural images. The second model is a slightly modified version of a model called A-ConvNets described in Chen et al. (2016). It is a 6-layer deep all-convolutional network specifically designed for analysing SAR-images. We modified the model slightly by adding three additional dropout-layers with a probability of 50% to randomly drop connections during training. This regularization techniques discourage neurons coupling and help to prevent overfitting during training, especially in presence of few data samples.

Two different inputs were tried for the VGG-19 as exemplified in Figure 1, namely [vv_2, vv_1, vv_2] and [vv_2, vv_1,(vh_2 - vh_1)]. For VGG-19 there were weights trained on millions of images from the ImageNet library (Deng et al., 2009) available. Here we followed three different approaches: (i) we trained the entire network from scratch with randomly initialised weights; (ii) we used the pre-trained weights

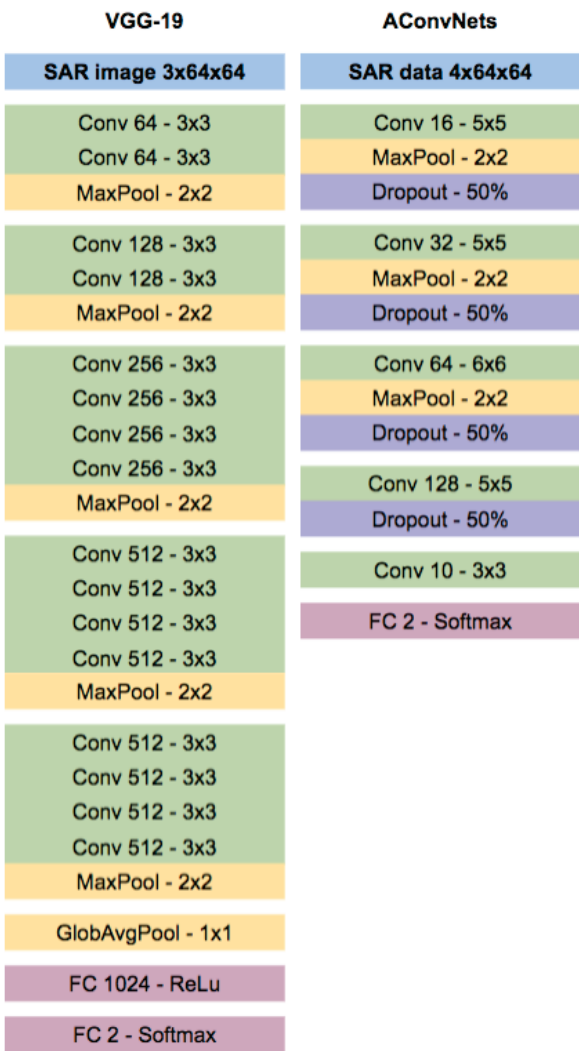

Figure 2: Design of the VGG-19 and AConvNets. Each layer in the models consists of multiple neurons The first active layer in the VGG-19 network is named “Conv 64 3x3”. This is a 64 neuron wide convolutional layer containing a 3x3 spatial filter. The various pooling layers shrink these spatial filters. The fully connected layers (FC) do not contain any spatial filters.

and froze the first ten layers; (iii) we used the pretrained weights and trained the entire network.

The procedure of ’recycling’ network parameters trained on different dataset is called ’transfer learning’ (Yosinski et al., 2014) and is grounded on the fact that the first layer in a CNN learns on low-level features (such as edges) that are general purpose and suitable for different applications. Indeed, it has been shown that CNN’s trained on different natural images mostly differ in the last layers (Razavian et al., 2014). Those are the ones that are fine-tuned. Transfer learning greatly helps to prevent overfitting in big networks when only few data are available. For transfer learning to be possible, the input data needs to be structured the same way. For the AConvNets we did not have available pretrained weights. We were not able to reuse pretrained weights since the SAR data also consisted of only four data channels. We encoded the input data as $[vv_1, vv_2, vh_1, vh_2]$ in this case.

For all experiments we used Python 3.5.2, Tensorflow 1.4.0 and Keras 2.0.9.

3. RESULTS

3.1. Accuracy assessment

The networks were trained to label the images in the two categories ’avalanche’ and ’no avalanche’. The reported accuracy is based on how well the networks performed when trying to predict the images in the validation dataset (Table 1). All training continued until the reported accuracy started to decline. The average of five runs is reported here. The training can often continue and receive close to perfect prediction of the training data, while the accuracy of the validation data stops or even declines. This is called overfitting, and will occur in neural networks when there is too little data available. At this stage it is reasonable to stop the training since further optimisation has no increased predictive capacity for data outside the training set.

3.2. Examples of CNN classification

Figure 3 illustrates 12 examples of slices classified by VGG-19 selected for illustrative purposes. The first six examples (1-6) are slices classified as ’avalanche’. Slices 1, 2 and 3 show avalanches that were easy to detect both for the expert and for the network. A qualitative assessment of the validation dataset indicates that the network makes few, if any errors in detecting these types of avalanches (e.g. 100% confidence). In slice 4 the network expresses 100% confidence that this is not an avalanche. The error stems from the avalanche being just outside the slice (lower right corner). The expert would not be able to classify this image as ’avalanche’ by just looking at the 64 x 64 pixels slice. The last two examples, 5 and 6, are examples of errors made by the network, which are accompanied by lower confidence levels. Avalanches 5 and 6 are diffuse avalanches that are also difficult for the expert to interpret.

In Figure 3 there are also six examples of the network classifying the slices as ’no avalanche’ (7-12). The first three examples (7-9) are correct (true negatives). Slices 10, 11 and 12 are examples of false positives where the network incorrectly classifies the slices as ’no avlanche’. There are areas with relatively high backscatter, however, insufficiently clear to be manually classified as ’avalanche’. From the errors it is relatively easy to understand that the network can be confused. The network seems to make errors in images that are also problematic to classify for humans.

Table 1: Accuracy of 5 different CNN configurations and their confidence interval (CI)

Net	Channels	Weights	Accuracy (95% CI)
VGG-19	vv_2, vv_1, vv_2	random	89.3% (88.2-90.5%)
VGG-19	vv_2,vv_1,(vh_2-vh_1)	random	90.7% (90.1-91.3%)
VGG-19	vv_2,vv_1,(vh_2-vh_1)	ImageNet	92.7% (91.8-93.5%)
VGG-19	vv_2,vv_1,(vh_2-vh_1)	ImageNet - 10 frozen	91.5% (90.6-92.4%)
AConvNets	vv_1,vv_2,vh_1,vh_2	random	90.8% (89.9-91.7%)

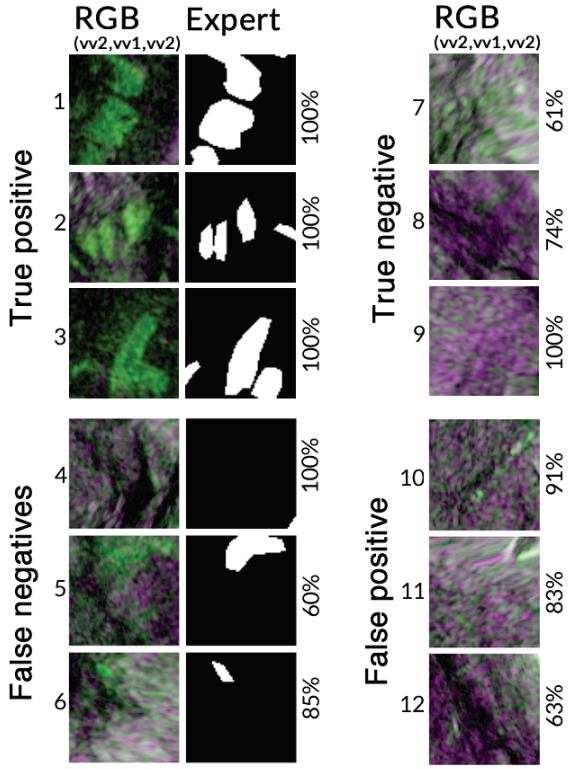

Figure 3: Examples of slices classified by VGG-19 and an expert (1-6) as well as false detections (7-12) The percentages indicate the network's confidence that the image is an avalanche (1-6) or that the image is not an avalanche (7-12).

4. DISCUSSION

Both CNN architectures considered in this work achieved promising results in detecting avalanches in SAR images, with accuracies around 90%.

The experiments were done on small 64 x 64 pixel slices, limiting the used networks to access a lot of the available information coming from the area surrounding an avalanche. Indeed, the manual identification of avalanches was done on the entire images, and a human interpreter would also have had major problems perfectly categorizing 64 x 64 pixel slices without being able to observe broader surroundings. We can therefore hypothesize that by including information on backscatter and slope angle from neighbouring slices, the networks performance would likely improve. This would be a next logical step to try to improve the network's accuracy.

The VGG-19 network was both trained using only VV polarisation [vv_2,vv_1,vv_2] as well as with adding VH polarisation to the third RGB channel [vv_2, vv_1, (vh_2 - vh_1)] (Figure 1). The version using only VV polarisation is identical to the image examined by the expert and achieves an accuracy of 89.3% (95% CI, 88.2-90.5%). Adding the VH polarisation on the other hand achieves an accuracy of 90.7% (95% CI, 90.1-91.3%). While the difference is not significant at an 0.05-level it is still worth noticing since adding additional data without any predictive effect usually should have caused a decline in accuracy. Furthermore this might be an indication that there is information about avalanches in the VH polarisation data that currently is not used.

We did see a significant effect of adding pre-trained weights. This was only possible for the VGG-19 network, and the best version did achieve an accuracy of 92.7% (95% CI, 91.8-93.5%). In convolutional networks so called low-level features are learned by the first layers, while high-level features are learned at later layers. In learning to recognise an image of a face, neurons recognising features like edges, curves and lines will fire in the first layers, while neurons recognising for instance an eye, will fire in one of the last layers.

It is worth mentioning that the pre-trained weights had been generated by training on more than a million of images from ImageNet, a collection of 'natural images' (Zoran, 2009). Natural images have a rich co-variance structure and their spectrum is usually non-Gaussian and follows a power-law distribution. It is not evident that this applies to the SAR-images and this potentially reduces the effect of transfer learning in our case. This hypothesis is also strengthened by the fact that there was no apparent effect of freezing the bottom layers. If the natural images and the SAR-images had the same underlying structure, this would usually have had a positive effect on the training. Similar research corroborates our findings (Penatti et al., 2015).

Concerning the training of model weights from scratch, a common rule-of-thumb is that few thousand of samples are required for each class in the dataset (Cireşan et al., 2012) to be able to completely fine tune the neural network. We did have 654 images of avalanches in the training dataset which is a bit less than is usually recommended. The fact that adding pre-trained weights improved accuracy, even when this weights are not ideally

suited, indicates that the result should improve when adding more labeled images.

While it is very time consuming to annotate thousands of avalanche images, an alternative could have been to pre-train our network on a large amount of other SAR-images and then fine-tuning it on the data for the specific task at hand. This alternative approach will be explored in future work.

In our experiments AConvNets performed about the same as the VGG-19 (90.8% (95% CI, 89.9-91.7%) vs 90.7% (95% CI, 90.1-91.3%)) when using randomly initialised weights. It was not possible in this project to add pre-trained weights for AConvNets, so it is hard to estimate the final potential of this network with the current amount of data. AConvNets has less than half the depth of VGG-19. Deeper networks are generally chosen when you need to remember complex high-level features, like it is often the case with natural images. That the same performance can be achieved with a simpler structure might indicate that learning high-level features is not an important property when analysing SAR-images. Simpler structure means shorter training time and networks that are easier to optimise, and should in general be chosen when giving similar performance.

5. CONCLUSION

To conclude, the experiments were mainly done to investigate the potential of using CNNs in detecting avalanches in SAR-images and if CNN detection is able to outperform conventional radar image classification. The obtained results are promising, especially since the networks have been trained end-to-end and without doing any pre-processing, apart from some masking of non-valid areas. Direct comparison of the accuracy between the tested CNN's and state-of-the-art image classification is limited by the different size of the slices used (64 x 64 pixels vs 50 x 50 pixels). However, the CNN's seem to produce consistently accuracy's of around 90%. It is likely that by using larger trainings datasets, accuracy can be improved even more. We have such datasets available to us from consistent automatic avalanche detection in Norway during the last two winters (2016-2018) (see companion paper from Eckerstorfer et al. in these proceedings).

ACKNOWLEDGEMENT

This work was partly financed by the Satskred 2.0 project, which is funded by the Norwegian Space Centre, the Norwegian Water Resources and Energy Directorate and the Norwegian Public Road Authorities.

REFERENCES

Chen, S., Wang, H., Xu, F., Jin, Y. Q. Target classification using the deep convolutional networks for SAR images *IEEE Transactions on Geoscience and Remote Sensing, 54(8)*. 4806-4817.

Cireşan, D. C., Meier, U., Schmidhuber, J. Transfer learning for Latin and Chinese characters with deep neural networks. *Neural Networks (IJCNN), IEEE, The 2012 International Joint Conference on Neural Networks*. 1-6.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L. A Large-Scale Hierarchical Image Database. *IEEE Computer Vision and Pattern Recognition (CVPR)*

Penatti, O.A., Nogueira, K., dos Santos, J.A. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? *Proceedings of the IEEE conference on computer vision and pattern recognition workshops.* 44-51.

Razavian, A.S., Azizpour, H., Sullivan, J., Carlsson, S. CNN features off-the-shelf: an astounding baseline for recognition *Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE.* 512–519.

Simonyan, K., Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv: 1409.1556*.

Vickers, H., Eckerstorfer, M., Malnes, E., Doulgeris, A. Synthetic Aperture Radar (SAR) Monitoring of Avalanche Activity: An Automated Detection Scheme in: Sharma, P., Bianchi, F.M., (Eds.) Image Analysis: 20th Scandinavian Conference, SCIA 2017, Tromsø, Norway. *Proceedings, Part II. Springer International Publishing, Cham.* 136-146.

Yosinski, J., Clune, J., Bengio, Y., Lipson, H. How transferable are features in deep neural networks? *Advances in neural information processing systems.* 3320-3328.

Zoran, D., Weiss, Y. Scale invariance and noise in natural images *Computer Vision, 2009 IEEE 12th International Conference on IEEE.* 2209-2216.