CONVOLUTIONAL AUTOENCODER FOR UNSUPERVISED REPRESENTATION LEARNING OF POLSAR TIME-SERIES

Thomas Di Martino[1,2]*, Regis Guinvarc'h*[1]*, Laetitia Thirion-Lefevre*[1]*, Elise Colin Koeniguer*[2]

[1] SONDRA, ONERA, CentraleSupélec, Université Paris-Saclay, 91190 Gif-sur-Yvette, France
[2] ONERA, Traitement de l'information et systèmes, Université Paris-Saclay, 91123 Palaiseau, France

ABSTRACT

Temporal Convolutional AutoEncoders are used as feature extractors to project time series onto a latent space where similarity detection can be easily performed. This model can generate accurate descriptors of the temporal profile of the input time-series. We apply this algorithm to PolSAR S1 uncoherent SAR time series where the model learns highly discriminative data representations. This reduction method is compared to others such as PCA or Temporal Averaging and is shown to outperform them when leveraging the learnt representation using K-Means clustering.

***Index Terms*—** Autoencoder, Deep Learning, dual polarization SAR, Time Series, Clustering

1. INTRODUCTION

When working with SAR interactions, different environments can be characterised by the value of their backscatter signal. The physical structures and dielectric characteristics of crops and forests, for example, are discriminating enough so that we can differentiate them by their backscatter signal value. It is also possible to discriminate among different forest types, as studied in [1].
However, crops and forests scattering properties evolve between seasons. For example, it was shown in [2] that the increase in VV and HH signal could be up to 10dB during the growth season. Such patterns signify that the average backscatter signal value may not always be enough to differentiate elements with similar physical structure: the temporal dimension of the signal, implying the use of multiple images, can be leveraged as a differentiating factor for this circumstance. The detection of trends and temporal profile of environments such as forests or crops have been explored for different data sources: for instance, with the use of Landsat Time Series for forest monitoring in [3] or for detection of forest disturbances with L-Band SAR images from ALOS and PALSAR in [4].
However, it has not always been a small matter to access Time Series of SAR images: with the advent of Big Data, the temporal analysis of the SAR backscatter for different environments is possible, to more considerable extent than it used to. New tools now allow researchers and users to access large preprocessed temporal stacks of intensity data from every corner of the Earth, which opens new opportunities to explore SAR Time Series.
The paper is organized in the following way: (2) presentation of the background surrounding Convolutional AutoEncoders; (3) description of the context, the task, and explored solutions; (4) presentation of our CAE architecture and of experimental results; (5) conclusions.

2. BACKGROUND

2.1. Nonlinear Principal Component Analysis

The concept of an Autoencoder was originally introduced in [5] where the author presents this architecture as a *Nonlinear Principal Component Analysis* (cf Fig.1).
Highly performant at eliminating nonlinear correlations within data, by learning an identity mapping with dimensionality reduction, the NLPCA or autoencoder consists of 3 main parts:

- The *Mapping* layer, or the encoder;
- The *Bottleneck* layer, or the latent representation layer;
- The *De-mapping* layer, or the decoder.

The former's task is to compress the input data into a representation with lower dimensions, called the input's "*features vector*" or "*embedding*". The decoder then uses this embedding to try to recreate the input data. Intuitively, assigning the task of the identity function to this network forces it to statistically learn a mapping with the least amount of correlated components. An assumption can then be that data with higher-level similarity (i.e. for time series, similarity in seasonality) will have similar embeddings.
These embeddings can then be used in tasks where a distance between two samples is computed, such as K-Means. As the generated embeddings contain a less noisy and more succinct version of the input, the computed distance becomes more reliable and tasks such as K-Means computation, which can see their performance decline in high dimensions, are improved.

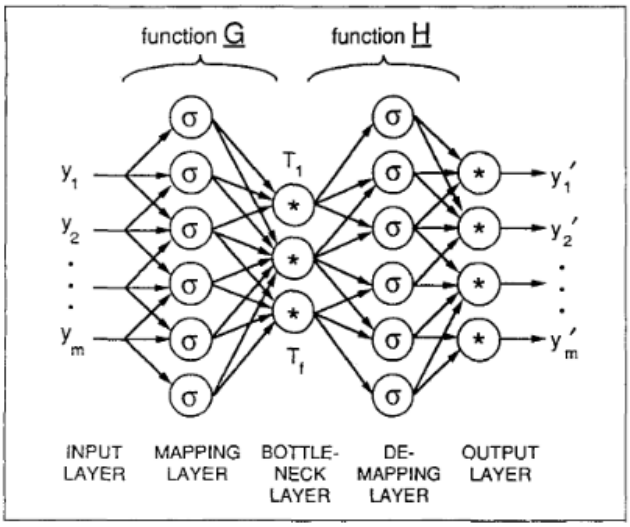

Fig. 1: Illustration of the NLPCA architecture (source: [5])

2.2. Convolutional Autoencoder

The example of Autoencoder presented in Fig. 1 used fully-connected layers, which can be used for data with no coherence between their features. However, in the case of Time Series, the temporal dimension needs to be accounted for. Two solutions are possible: we can either use Recurrent Neural Nets or 1D Convolutional Neural Nets. However, we choose the latter as empirically found that the reconstruction capabilities of 1D CNNs exceed their recurrent peers.

1D Convolutional Autoencoders (CAE) use convolutional layers as feature extractors and exploit the known correlation between neighbouring temporal features. The encoder part of a 1D-CAE will reduce the input's temporal dimension but will use a high number of filters. Once flattened, the high-dimensional feature vector is mapped using a traditional fully connected layer to the embedding layer. Oppositely, a similarly sized vector is recreated from the embedding layer using a fully connected layer before being fed to the decoder. The decoder, responsible for reconstructing the input signal, uses "*Deconvolution*", or transpose convolution, for dimension expansion.

Once reconstructed, the time series is compared against its original input using a Mean-Squared-Error loss function. Other works have used multi-task learning to improve the embedding, such as in [6] where they use an auxiliary truncated-SVD during training to model the K-Means objective directly within the autoencoder training pipeline.

3. CONVOLUTIONAL AUTOENCODER FOR SAR TIME-SERIES

Convolutional Autoencoders have already been used to extract embeddings from SAR Time Series as in [7] where the authors use 3D Convolutions, exploiting both temporal and spatial dimension (resp. 1D & 2D) to represent their input data as an encoded feature vector, with the purpose of clustering. For that matter, they represent each pixel $p_{(i,j)}$ as a neighbouring patch of odd size s, to model its neighbourhood. This added information can better define each pixel $p_{(i,j)}$ with regards to its environment, but it may also lead to information leakage & redundancy in case of two pixels sharing neighbourhood. For that matter, this could diminish the contribution of the temporal dimension within the embedding.

Hence we choose to focus on a fully-temporal architecture, relying on no spatial information to create an embedding solely based on the temporal profile of a pixel.

Given a temporal stack of T dual-pol SAR images consisting of N pixels, we transform it into a list of time-series $l = \{p_i, \forall i \in [\![1, N]\!]\}$ where each p_i is a multimodal time-series such that:

$$p_i = [p_i^{(1)}, p_i^{(2)}, ..., p_i^{(T)}], p_i^{(t)} = (VV_i^{(t)}, VH_i^{(t)})$$

In this paper, we investigate candidates for an optimal function $f : \mathbb{R}^{T\times 2} \to \mathbb{R}^2$ that projects the fore-mentioned time-series into a latent space where similarity computation and clustering can be performed more quickly and efficiently. This function is expected to use seasonality, inter-annual and intra-annual trends as leverages for its low-dimension representation task. Multiple candidates are kept for comparison:

- Temporal Averaging: $f(p) = \frac{1}{T} * \sum_{t=1}^{T} p_.^{(t)}$;
- PCA: $f(p) = flatten(p) * M$ where M is a 2-columns matrix consisting of the two first principal components of a PCA of l;
- CAE: $f(p) = encoder(p)$ where the encoder function is trained in a CAE using a reconstruction task.

The details about the CAE's architecture and experiments are presented in the following section.

4. EXPERIMENTAL RESULTS

4.1. Architecture

As presented in Fig.2, the multimodal SAR time-series are fed to a CAE with the parameters presented in table.1. We use two Convolutional layers to extract features that are then mapped onto a 2-sized vector using fully connected layers. This vector is then transformed back into the original time series using transposed convolutions, after which one last convolution aims at cleaning their potentially rough output.

In our experiments, we worked with a stack of $T = 189$ dates, with recordings from Jan. 2019 to Nov. 2020, with a total of around $N = 4e5$ time series. Only images with an ascending orbit during acquisition were kept. The training was run for 100 epochs using ADAM Optimiser and a learning rate of

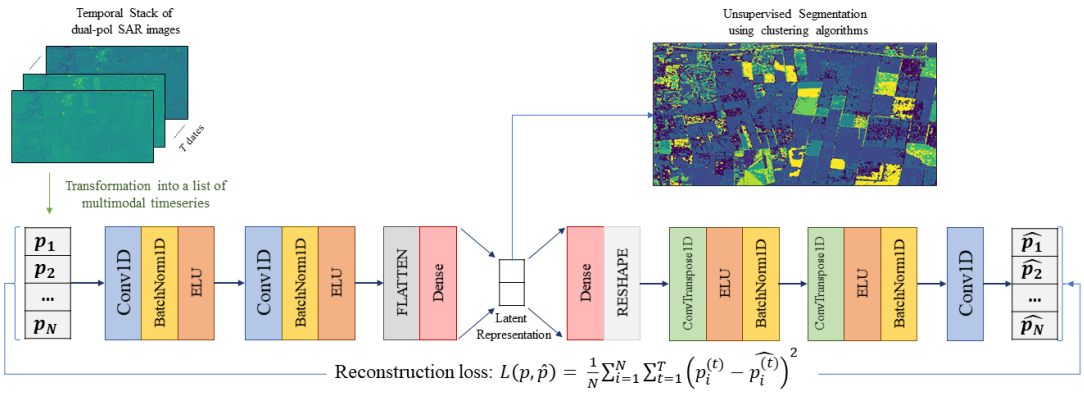

Fig. 2: Temporal Convolutional Autoencoder model

Table 1: Parameters of our CAE architecture

#1	Conv1D	64 Filters, Kernel Size 3, Stride 2
#2	Conv1D	128 Filters, Kernel Size 5, Stride 2
#3	Dense	(5760,100)
#4	Dense (Embedding)	(100,2)
#5	Dense	(2,5760)
#6	ConvTranspose1D	64 Filters, Kernel Size 5, Stride 2
#7	ConvTranspose1D	2 Filters, Kernel Size 3, Stride 2
#8	Conv1D	2 Filters, Kernel Size 3, Stride 2

$5e-3$. The training batch size was set to 1024 with a train/test split of 80/20. On an RTX 3090, one epoch lasts 30sec. Experiments were divided into two phases:

1. Train the autoencoder using a reconstruction task;
2. Group each time series' embedding with K-Means;

4.2. Results

As presented in Fig.3, we experiment with PCA decomposition, Temporal Averaging (TA), and a CAE architecture. The comparison between the CAE and the TA intends to demonstrate the value of using the temporal dimension to analyse land cover for tasks such as unsupervised segmentation using clustering algorithms. The second, between PCA and CAE, is to justify the need to explicitly model the temporal interaction, rather than use a vectorised representation of the data. This temporal concept is leveraged with the use of 1D Convolutional Autoencoders for dimension reduction. Each of these methods reduces pixels' multimodal time series to a vector of 2 values. They are being evaluated on their degree of expressiveness and discriminability with regards to the original time series.

In Fig.3, we compare the components of each 2-sized vector (TA, PCA, and CAE): we notice a higher contrast for the deep embeddings of pixels than for the other methods. This

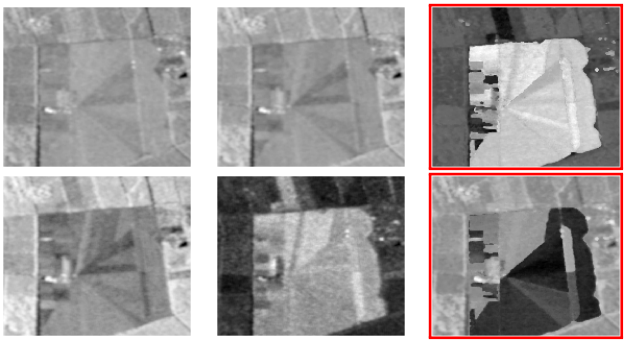

Fig. 3: Visual comparison of algorithms applied to Forêt Nézer

high contrast, here between a farm and the surrounding trees, let us presume higher semantic separability between environments with different temporal behaviour when using the CAEs as descriptors. To verify this visual observation, we run a K-Means, with $K = 2$, algorithm using the data representation from each of these three methods. We then check both the spatial and temporal coherence of the clusters.

K-Means algorithm results quality is a consequence of an already existing separability potential within data representation. When observing Fig. 4, we notice a higher degree of divisibility between CAE embeddings. While the 2nd component of the PCA plots appears to offer better separability than the VH TA, it still is more cluttered than vectors from the CAE. We can suppose that the two sets of spikes in the histograms of the CAE components correspond to a distinction between the area of the farm and its surroundings, displaying a higher spatial coherence than with TA or PCA.

When running K-Means over the resulting vectors, we indeed notice, in Fig.5, a much clearer distinction between the farm environment and its surroundings than with other methods. The farm outline's unusual shape on the right side of the image, consisting of round corners, is preserved when us-

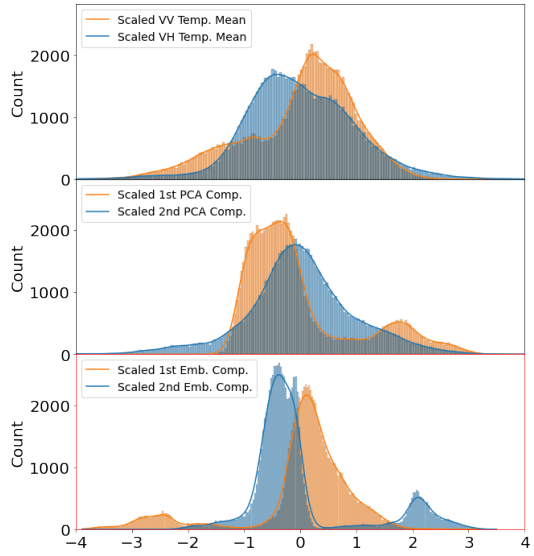

Fig. 4: Histograms of each 3 methods (TA, PCA, CAE)

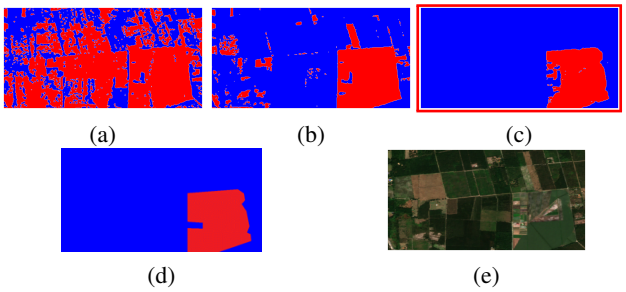

Fig. 5: Binary K-Means for TA (a), PCA (b) and CAE (c) components (Red: Farm, Blue: Other) compared to ground truth label (d) and Sentinel 2 B4/B3/B2 RGB imagery (e)

ing the CAE to encode the time series. When plotting the histogram, using the cluster attribute as a colour code, we observe that each cluster indeed corresponds to specific spikes amongst the histogram of Fig.4. This shows that CAE architectures can discriminate various environments based on their temporal profile, within which lies a lot of physical information about the surface. In table. 2, we display the F1 score as a

Table 2: F1 Score evaluation of each method's unsupervised clustering performance

Evaluation	TA	PCA	CAE
F1 Score	0.39	0.78	0.94

criteria of quantitative performance for each algorithm, tasked with the retrieva of the farm area without supervision, using the k-Means algorithm over the computed embeddings. As initially presented in Fig.5, we have a noticeable superiority of the CAE algorithm over the two other methods, F1-score wise.

5. CONCLUSION

In this paper, we investigate the potential of Convolutional Autoencoders (CAE) for dimension reduction of Time Series while retaining information related to trends, inter or intra-annual seasonalities in PolSAR S1 Time Series. Its performance is evaluated on a farm located next to Forêt Nézer. We compare it to using the Temporal Average (TA) or a PCA for data representation of the time series. The CAE representational performance is shown to be able to discriminate the farm's temporal profile against its surroundings. At this task, the TA and the PCA are shown to have much less potential at representing time-dependent features and output a messier result. While applied to an agricultural environment, the CAE can also be leveraged to detect and discriminate naturally occurring seasonalities, amongst forests for example.

6. REFERENCES

[1] A. Lapini, S. Pettinato, E. Santi, S. Paloscia, G. Fontanelli, and A. Garzelli, "Comparison of machine learning methods applied to sar images for forest classification in mediterranean areas," *Remote Sensing*, vol. 12, no. 3, pp. 369, Jan 2020.

[2] T. Le Toan, F. Ribbes, Li-Fang Wang, N. Floury, Kung-Hau Ding, Jin Au Kong, M. Fujita, and T. Kurosu, "Rice crop mapping and monitoring using ers-1 data based on experiment and modeling results," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 35, no. 1, pp. 41–56, 1997.

[3] R. E. Kennedy, Z. Yang, and W. B. Cohen, "Detecting trends in forest disturbance and recovery using yearly landsat time series: 1. landtrendr — temporal segmentation algorithms," *Remote Sensing of Environment*, vol. 114, no. 12, pp. 2897–2910, Dec 2010.

[4] C. Marshak, M. Simard, and M. Denbina, "Object-oriented monitoring of forest disturbances with alos/palsar time-series," in *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, 2019, pp. 1629–1632.

[5] M. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *Aiche Journal*, vol. 37, pp. 233–243, 1991.

[6] Q. Ma, J. Zheng, S. Li, and G. W. Cottrell, "Learning representations for time series clustering," in *Advances in Neural Information Processing Systems*, 2019, vol. 32, p. 3781–3791.

[7] E. Kalinicheva, J. Sublime, and M. Trocan, "Unsupervised satellite image time series clustering using object-based approaches and 3d convolutional autoencoder," *Remote Sensing*, vol. 12, no. 11, pp. 1816, Jan 2020.