Ultrasonic signal noise reduction based on convolutional autoencoders for NDT applications

Nabil Jai Mansouri[1,2], *Ghizlane* Khaissidi[2], *Gilles* Despaux[1], *Mostafa* Mrabti[2], and *Emmanuel* Le Clézio[1]

[1]IES, Univ Montpellier, CNRS, Montpellier, France
[2]FST, Sidi Mohamed Ben Abdellah University, Fez, Morocco

Abstract. One of the most challenging problems of ultrasonic non-destructive testing is the signal distortion caused by the presence of noise, yielding the sound wave corruption and thus degrading the ultrasonic imaging technology performance due to Time of flight methods' loss of precision. Deep learning algorithms have proven their effectiveness in reducing noise on several types of signals in different domains. In this paper, we propose a one-dimensional convolutional autoencoder for ultrasonic signal denoising. The efficiency of the proposed architecture is compared to the wavelet decomposition method, collating the peak signal-to-noise ratio values on the denoised signals. Our method proved its potential for NDT applications in recovering temporal information even on very noisy signals, and improving the PSNR by about 30 dB.

1 Introduction

Noise reduction is one of the major issues in the domain of industrial or medical ultrasonic imaging. Indeed, improving the Peak Signal-to-Noise Ratio (PSNR) generally allows early detection of modifications in the tested media. In the non-destructive testing field, noise reduction is performed on ultrasonic signals using several digital signal processing techniques. These methods categorize the noise into two clusters. In the first case, the noise has different frequency properties compared to the sound waves generated by the transducer and could be filtered using elementary filtering methods. In the second case, the noise shares the same frequency band as the ultrasound waves, principally caused by the heterogeneity of the materials on which the discontinuities result in echoes sharing the same properties as the sound waves emitted by the transducer. Therefore, alternative and advanced denoising methods were proposed considering the time-domain information, the frequency-domain information, or both. Effective noise reduction methods comprise split spectrum processing SSP [1-2], wavelet transform methods [3-4], and learning-based algorithms [5-6].

Deep learning (DL) in parallel to computation power improvement has lately gained researchers' attention. It offers the possibility to model architectures enabling the extraction of high complexity features, which permit solving sophisticated problems as noise reduction. It has proven its effectiveness and has been used for signals, images, and video denoising. Early work on ultrasonic signal denoising used artificial neural networks and proved their effectiveness compared to singular value decomposition (SVD), wavelet algorithms, and principal component analysis (PCA) [7]. Further work aimed to suppress grain noise on ultrasonic signals by clustering correlative signals before imputing them into an autoencoder [8].

In this paper, convolutional autoencoders (CAE) are used for noise reduction on ultrasonic signals. In Section 2, we present basic ideas of convolutional neural networks and autoencoders along with a description of the signals on which they were trained. In section 3 we discuss the results of the proposed method and compare its performance to discrete wavelet transform algorithm collating the after noise reduction's peak signal-to-noise ratio of these methods and comparing the resulting signal traces.

2 Methodology

Deep learning is a subset of machine learning that resurged lately thanks to the advancement of the algorithms along with the improvement of the computing ship's power especially GPUs and TPUs [9]. Particularly, convolutional neural networks have proven their performance extracting locally connected features. This variety of neural networks are formed by three types of layers: convolutional layers, pooling layers, and fully connected layers. Firstly, filters of the convolutional layers are convolved to the input in a sliding window resulting in a feature map summarizing the presence of detected features in the input. The convolution of each feature map's element in the following layer is calculated as donated in formula (1):

$$\mathbf{S}(i,j) = (\mathbf{I} * \mathbf{K})(i,j) = \sum_m \sum_n \mathbf{I}(m,n)\mathbf{K}(i-m, j-n) \quad (1)$$

where $\mathbf{S}(i,j)$ donate the feature maps, $\mathbf{I}$ the input, and $\mathbf{K}$ the kernel.

The feature maps are processed through an activation layer as rectified linear unit (ReLU), exponential linear unit (ELU), leaky ReLU, hyperbolic tangent (tanh), etc. The complexity of the learned features increases across the depth of the layers. The following layer is usually a pooling layer, frequently a Max-pooling layer, that reduces the computational costs as it ensures the shift-invariance which is a common feature of the architecture. Depending on the applications of the DL model, the previous layers are followed by a fully connected layer for a classification or regression task. In case an overfitting phenomenon is noticed, several methods of regularization could be employed. A widely used regularization method is Dropout [10].

On the other hand, autoencoders are the most common architecture used for noise reduction purposes. As shown in Fig. 1, they learn an identity function reconstructing the clean version of its input in the outputs by tuning the encoder and decoder fully connected layer's parameters, minimizing the reconstruction error. It starts by compacting the input into a latent space, which is leveraged to reconstruct the input at the output during the decompression process.

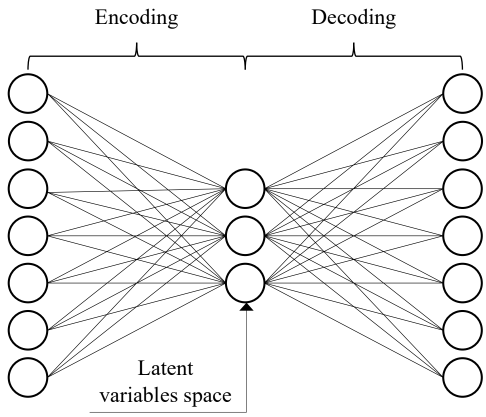

Fig. 1. Autoencoder's architecture

2.1 Convolutional autoencoder

In this work, we combined the feature's extraction performance of the CNN with the autoencoder denoising efficiency. In contrast to the traditional autoencoders, the proposed CAE's encoder contains convolutional layers, and decoder contains transposed convolution layers. The down-sampling of the inputs is ensured by sliding the filters for more than one element. The number of elements skipped is called a stride. Same for the transposed convolution, strides are employed to up-sample these layers' inputs.

The CAE architecture is designed carefully in a manner that the model's input shape will be restored in its output. The hyperparameters of the model were tuned in three steps. First, a Random Search on wide space of hyperparameters, followed by a Grid Search around the optimum resulted from the first step and then tuned manually collating the model denoising performance on each iteration. We noticed that the depth of the CAE varied proportionally to the noise reduction, although it ensued over-fitting that was resolved using the dropout regularization method at each layer, where the dropout rate was fixed to 0.1. ReLU activations were used for sparsity and vanishing gradient prevention. The CAE's parameters were optimized using the adaptative moment estimation (ADAM) algorithm [11] minimizing the peak signal-to-noise ratio (PSNR) as a loss function. The ADAM optimizer has been used as it prevents premature convergence to local minima. This is achieved thanks to its adaptative learning rate. This optimizer's property speed-up the learning as well, resulting in faster convergence. The CAE was created under TensorFlow and trained on Nvidia's GPU Tesla K80. The architecture of the proposed CAE is shown in Fig. 2.

2.2 Dataset

To train our CAE, we created a database of 100,000 ultrasonic signals, simulating a pulse-echo ultrasonic non-destructive testing operation. The used transducer frequency was 20 MHz, and signals contained 1,000 datapoints sampled with a 5 ns rate. The pulses and echoes were synthesized at different delays for each database signal. We added Gaussian white noise to these signals to have 3 different levels of noise intensities: 10 dB, 20 dB, and 30 dB, uniformly distributed. We trained, tested, and validated our CAE on 90 %, 5 %, and 5 % of the dataset samples, respectively.

3 Results

To evaluate the noise reduction performance of the proposed method, we created 3 databases of 50 samples each. The first, second, and third databases contained noisy signals at 10 dB, 20 dB, and 30 dB noise intensities, respectively. Examples of each database are shown in Fig. 3. We denoised these databases using our CAE and wavelets decomposition method, and then compared the PSNR results of the two methods. Considering the wavelet decomposition method, we decomposed the signals using the 'sym2' basis function, into the maximum levels in a sort to prevent the coefficients' corruption [12]. These coefficients were thresholded using the Bayes shrink algorithm [13]. As shown in Fig. 4, we visualize the resulting waveform of a sample from each database for better analysis of the method's behavior.

Figure 4 presents in columns (1) the noisy initial signals, (2) the wavelet decomposition results, and (3) the outputs of the CAE. They are compared to the un-noisy reference signal. As shown in Fig. 4 (a, 2), we noticed that the wavelets decomposition method fails the peak reconstruction, as it did not reduce significatively the noise on signals which PSNR values before denoising were estimated to 10 dB. The noise reduction is better performed on the signals from the second database shown in Fig. 4 (b, 2), and greater on the third one presented in Fig. 4 (c, 2). Considering, the peaks recovery on these databases samples, we observed that the echoes were distinguished, but moderately distorted due to the grain noise suppression. On the other hand, the proposed CAE

https://doi.org/10.1051/e3sconf/202235101039

succeeded in the peaks reconstruction, even on signals with low PSNR values.

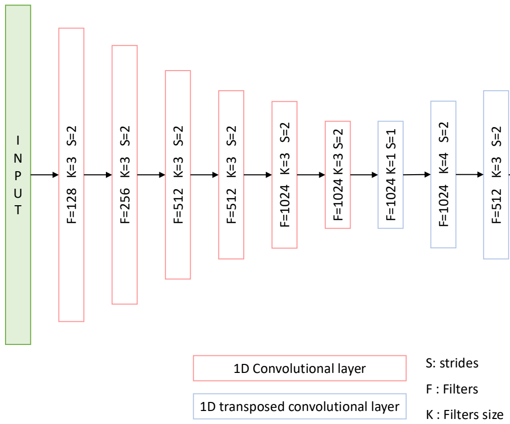

Fig. 2. The proposed CAE architecture.

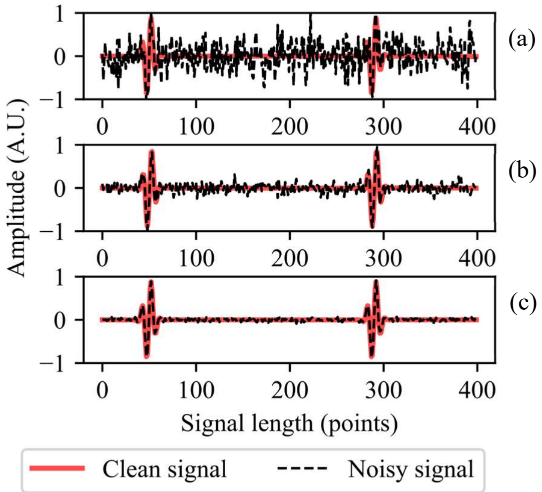

Fig. 3. Sample of the (a) 10 dB, (b) 20 dB , and (c) 30 dB SNR databases.

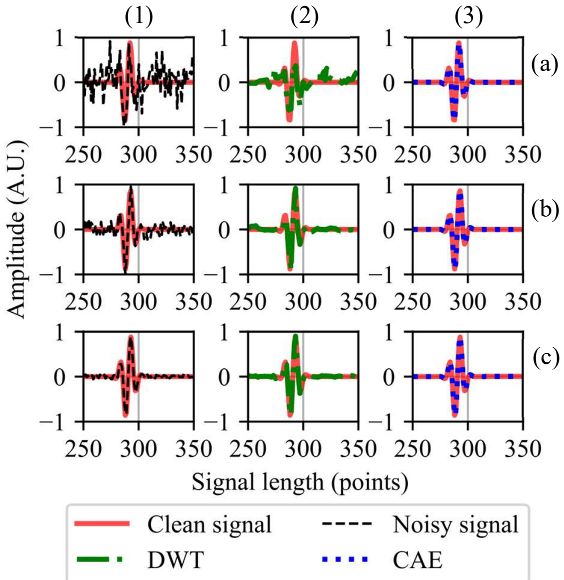

Fig. 4. Sound waves before and after denoising using CAE and wavelets decomposition for the three intensity levels of each database. (a) 10 dB, (b) 20 dB , and (c) 30 dB PSNR.

Considering noise suppression, it is noticeable that the proposed method was able, whatever the initial noise level, to learn the features of the sound waves, ensuring an efficient dissociation of the Gaussian white noise from the ultrasonic signal. At the level of the visual comparison, our CAE clearly outperforms the wavelet method, reconstructing the denoised signal whatever the initial PSNR (Fig. 4.(3)).

To strengthen the previous analysis, we compared the PSNR enhancement obtained by the two methods after denoising. To this end, we calculated the mean value of the signal's PSNR distribution of each database after the noise reduction.

As presented in Fig. 5, the wavelet decomposition method performed a ≈12 dB PSNR improvement on signals from the first database, compared to the ≈30 dB enhancement performed by our CAE. Considering signals from the second and third databases, the wavelets method moderately improved the PSNR by ≈7 dB while our CAE conserved its performance and augmented the signals PSNR by ≈28 dB. As a conclusion, whatever the initial PSNR, the proposed CAE yields signal denoising with PSNR about 20 dB better than the wavelet method.

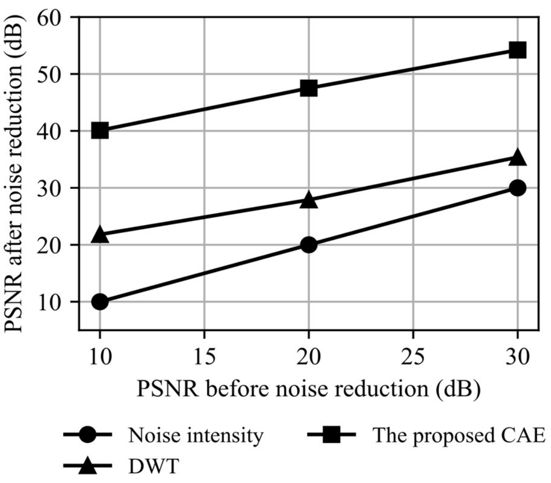

Fig. 5. Mean of PSNR values of each database sample after denoising with the proposed CAE and wavelets decomposition method.

4 Summary

In this work, we proposed a convolutional autoencoder for ultrasonic signal denoising. Our method proved its efficiency in recovering the temporal information even on very noisy signals, reconstructing the sound waves, and improving the PSNR by about 30 dB. These accomplishments push further the limits of the ultrasonic NDT methods and enables deep exploration despites low signals' PSNR. Ultrasonic signals denoising using convolutional autoencoders requires tuning of the hyperparameters of the model (Number of layers, Number and size of filters, regularizations, activations,..) for other signal distributions. Pre-trained models could be recycled for similar application's use utilizing transfer learning techniques [14], reducing costs and time consumption of the training process. Future work will deal with the application of this CAE on experimental signals in a way to enhance the ultrasonic non-destructive testing performance on complex materials and allow the high-quality control of their heterogeneities.

References

1. S. K. Pedram, S. Fateri, L. Gan, A. Haig, and K. Thornicroft, "Split-spectrum processing technique for SNR enhancement of ultrasonic guided wave," *Ultrasonics*, **83**, pp. 48–59, (2018)
2. P. M. Shankar, V. L. Newhouse, P. Karpur, and J. L. Rose, "Split-Spectrum Processing: Analysis of Polarity Thresholding Algorithm for Improvement of Signal-to-Noise Ratio and Detectability in Ultrasonic Signals," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, **36**, pp. 101–108, (1989)
3. E. Pardo, J. L. San Emeterio, M. A. Rodriguez, and A. Ramos, "Noise reduction in ultrasonic NDT using undecimated wavelet transforms," *Ultrasonics*, **44**, pp. 1063–1067, (2006)
4. A. Abbate, J. Koay, J. Frankel, S. C. Schroeder, and P. Das, "Signal detection and noise suppression using a wavelet transform signal processor: application to ultrasonic flaw detection," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, **44**, pp. 14–26, (1997)
5. R. Vicen, R. Gil, P. Jarabo, M. Rosa, F. López, and D. Martínez, "Non-linear filtering of ultrasonic signals using neural networks," *Ultrasonics*, **42**, pp. 355–360, (2004)
6. A. Chapon, D. Pereira, M. Toews, and P. Belanger, "Deconvolution of ultrasonic signals using a convolutional neural network," *Ultrasonics*, **111**, p. 106312, (2021)
7. F. Gao, B. Li, L. Chen, X. Wei, Z. Shang, and C. He, "Ultrasonic signal denoising based on autoencoder," *Review of Scientific Instruments*, **91**, (2020)
8. W. Xu, X. Li, J. Zhang, Z. Xue, and J. Cao, "Ultrasonic signal enhancement for coarse grain materials by machine learning analysis," *Ultrasonics*, **117**, p. 106550, (2021)
9. Y. E. Wang, G.-Y. Wei, and D. Brooks, "Benchmarking TPU, GPU, and CPU Platforms for Deep Learning," (July 2019)
10. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal of Machine Learning Research*, **15**, pp. 1929–1958, (January 2014)
11. D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–15, (2015)
12. S. G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Transactions on Image Processing*, **9**, pp. 1532–1546, (2000)
13. D. L. Donoho and J. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, **81**, pp. 425–455, (1994)
14. S. Panigrahi, A. Nanda, and T. Swarnkar, "A Survey on Transfer Learning," *Smart Innovation, Systems and Technologies*, **194**, pp. 781–789, (2021)