Denoising Induction Motor Sounds Using an Autoencoder

Thanh Tran†, Sebastian Bader*, Jan Lundgren*
* STC Research Centre, Department of Computer and Electrical Engineering, Mid Sweden University
† Group Technology Team, Assa Abloy, Stockholm, Sweden

thanhtn02@gmail.com, {thanh.tran, sebastian.bader, jan.lundgren}@miun.se

Abstract—Denoising sound is essential for improving signal quality in various applications such as speech processing, sound event classification, and machine failure detection systems. This paper proposes an autoencoder method to remove two types of noise, Gaussian white noise, and environmental noise from water flow, from induction motor sounds. The method is trained and evaluated on a dataset of 246 sounds from the Machinery Fault Database (MAFAULDA). The denoising effectiveness is measured using the mean square error (MSE), which indicates that both noise types can be significantly reduced with the proposed method. The MSE is below or equal to 0.15 for normal operation sounds and misalignment sounds. This improvement in signal quality can facilitate further processing, such as induction motor operation classification. Overall, this work presents a promising approach for denoising machine sounds using an autoencoder, with potential for application in other industrial settings.

Index Terms—Autoencoder, Denoise sound.

I. INTRODUCTION

In the manufacturing industry, machine failure detection systems are critical to automatically detect broken components in machines for scheduled maintenance [1], [2]. This can minimize the downtime of the machine, and thus the maintenance cost. Researchers have recently focused on exploiting emitted sounds to detect machine failures. Noise, however, can be present in the recorded sound signals of machines in factories. The noise can originate from sources such as people talking, flowing water, wind, or other machines in the background. If the noise of the sound is removed, the quality of the recorded sound event will increase, resulting in the successful application of machine failure detection systems based on the sound signals. In this paper, a method is investigated for denoising machine sounds by mapping noisy sounds to clean ones using an autoencoder.

For many years, there has been a growth of methods to reduce noise in sounds and speech. These methods include approaches such as minimum mean square error (MMSE) [3], spectral subtraction [4]–[7], Wiener filtering [8]–[10], time-frequency block threshold [11], nearest neighbor estimation [12], Singular Value Decomposition (SVD) [13], and double-density dual-tree discrete wavelet transform (DDDTWT) [14].

Although these methods have demonstrated good performance in image, sound, and speech denoising, they still have some drawbacks. They are only effective at low levels of noise and are based on assumptions of aggregate statistical models. Hence, these algorithms frequently underestimate or overestimate noise, which results in either insufficient noise removal or audio distortions caused by excessive noise removal. Moreover, identifying the gain of a Wiener filter requires knowledge of the power spectral densities (PSDs) of the noise and the desired signals at a particular frequency. The SVD computation is slow and has a high computational cost. The choice of the proper wavelets for denoising purposes using wavelet transform can be time-consuming [15].

Using deep neural networks (DNNs) as a denoising method can address these problems. DNN-based methods use pairs of noisy sounds and their corresponding clean sounds to be trained. This approach has been shown to outperform conventional denoising filters for images [16] and time-series signals [17]–[22]. Specifically, deep learning can effectively remove the issue of Gaussian noise in sounds. In supervised learning approaches, pre-trained DNNs combined with linear support vector machines have been used to learn and predict features of noisy signals [23]. More recently, convolutional neural networks (CNNs) are attracting considerable interest in many applications in signal processing, image processing, and computer vision. CNNs have been demonstrated to enhance a large number of noisy images with different standard noise, thus allowing for a greater degree of generalization [24]. Some hybrid methods that are the combination between conventional denoising methods and deep learning methods have also been proposed to denoise sound signals [25]–[27].

Autoencoders have been applied to reduce noise in images [28]–[32], and have shown better performance than conventional noise filters. A reason for this is that autoencoders can be modified based on the input images, and hence could be labeled to be data-specific. For denoising images, images are corrupted by adding random noise, and the autoencoder is trained on the original images and the corresponding noisy images (corrupted images) to produce noise-free images (uncorrupted images). Recently, denoising autoencoders have also been used successfully to denoise ECG signal [33]–[35] and radio signals [36]–[38].

The noise in sound is frequently assumed to follow a Gaussian distribution in conventional denoising algorithms. Real noise in sound, however, is more complicated. Machine sounds in factories, for example, are rarely particular to the machine
and may include ambient noise such as human-produced sounds or sounds from other machines. The spectral content of background noise and the sound event can sometimes overlap. Conventional denoising approaches may not perform well in these situations because they may remove portions of the sound event that sound like background noise. Due to this, there may be audible distortions as a result of the denoising process. Hence, effective denoising technologies that can eliminate various types of environmental noise (background noise) without sound distortion are still needed.

The autoencoder’s effectiveness in other applications motivates its use in sound signal denoising, particularly for the removal of random white noise and various types of acoustic noise in real life. As a result, the goal of this research is to develop, implement, and evaluate an autoencoder for the reduction of noise to improve the quality of sound events. We attempt to denoise sounds to become as close to their original clean sound as possible by learning the noise distribution in sounds and denoising them. In the scope of this paper, two types of noise were manually added to the original clean sound signals of industrial motors. The autoencoder is then trained using the original clean sound and the corresponding noisy sound. The autoencoder recovers the noiseless sound from the noisy signal. To our knowledge, this is the first study to evaluate a denoising autoencoder for noise reduction in industrial motor sounds.

The rest of this paper is organized as follows. A brief overview of the utilized dataset is presented in Section II. The proposed method is described in Section III. The evaluation results are presented in Section IV. In Section V and Section VI, we discuss the obtained results and draw conclusions, respectively.

II. DATASET

For the investigation in this study, signals from MAFAULDA are used. The MAFAULDA dataset [39] includes time series data that are acquired from the "SpectraQuest’s Machinery Fault Simulator (MFS) Alignment-Balance-Vibration (ABVT)" using different sensors, including microphones. The data acquisition system includes three industrial vibration monitoring instrumentations (IMI sensors, model 601A01 accelerometers), one triaxial accelerometer (model 604B31), Monarch Instrument MT-190 analog tachometer, Shure SM81 microphone, and two National Instruments NI 9234 4 channel analog acquisition modules with a sample rate of 51,200 Hz. The dataset is divided into six categories: normal function, imbalance faults, vertical misalignment faults, horizontal misalignment faults, inner bearing faults, and outer bearing faults. Due to the application aspect in detecting and classifying industrial motor faults, this dataset was selected for this research. For our investigation, we focused on signals from the normal function and horizontal misalignment fault categories. There are 49 sounds in the normal function category when the machine is working properly. There are 197 sounds in the horizontal misalignment category, divided into four types: 0.5mm misalignment (50 sounds), 1.0mm misalignment (49 sounds), 1.5mm misalignment (49 sounds), and 2.0mm misalignment (49 sounds).

The sound signals were recorded by Shure SM81 microphones with a sample rate of 50,000 Hz and stored in comma-separated values (CSV) files. Before further analysis, the CSV files were converted into waveform audio file format (WAV). Each signal was sampled at a rate of 50,000 Hz for 5 seconds, resulting in time series of 250,000 samples.

III. METHODOLOGY

An encoder and a decoder are the two halves of an autoencoder. The encoder compresses the data that it receives. The decoder then uses the encoder’s latent representation to reconstruct the output. To train the denoising model, noisy sounds and their corresponding clean sounds were fed into an autoencoder. The encoder learns the noisy sounds during the encoding process, while the decoder attempts to generate new sounds that best match the clean input sounds.

As a first step, the noise was added to the original induction motor sounds. In this study, two types of noise were added: a random white Gaussian noise and water sink faucet noise.

The random noise was generated stochastically from a Gaussian distribution where 0 is mean and 1 is the standard deviation. The length of each random noise was equal to the length of the sound in the dataset. This noise is multiplied by the noise factor of 0.1 and then added to the original sounds to create corresponding noisy sounds. Hence, each sound in the dataset was superimposed with a different random noise.

Environmental noise can be classified into pink noise (e.g., heartbeats, steady rain, wind, waves), brown noise (thunder), and blue noise (water from a faucet). The noise of flowing water is common in many factories due to its usage in processing, cleaning, diluting, or cooling. Hence, a denoising autoencoder was trained with a specific type of noise (the sound of the water sink faucet as an example). The water sink faucet sound is blue noise. The spectral density (power per hertz) of blue noise is proportional to its frequency. As the frequency increases, the energy and power of the signal increase as well. The water sink faucet noise is 5 seconds in length, the same as the length of the original sound in the dataset.

The encoder consists of four 1D-convolutional layers with a kernel size of 3. The first 1D-convolutional layer has a filter size of 128 and a rectifier linear unit (ReLU) as its activation function. The input size of the encoder is (250,000, 1), corresponding to 250,000 samples in each time series. The second, third, and fourth 1D-convolutional layers have filter sizes of 32, 16, and 8, respectively. Similarly, the decoder consists of four 1D-convolutional transpose layers and a 1D-convolutional layer. The first, second, third, and fourth 1D-convolutional transpose layers have filter sizes of 8, 16, 32, and 128, respectively. The final convolutional layer turns the number of channels back into one. The batch size is 8 and the number of epochs is 10. We used 30% of sounds for testing, 56% for training, and 14% for validation. The max
norm constraints regularization was utilized with a value of 2.0 so the updates of the network are always bounded. Using projected gradient descent to enforce the constraint, maximum norm constraints establish an absolute upper limit on each neuron’s weight vector. The binary cross-entropy was utilized as the loss function. Hence, the original sounds and noisy sounds were normalized so that their values are in the range of [0, 1] using

\[ n_i = (y_i - \min(y))/(\max(y) - \min(y)), \]  

with

- \( n_i \): the normalized value \( i_{th} \) in the dataset;
- \( y_i \): the sample value \( i_{th} \) in the dataset;
- \( \min(y) \): the minimum value in the dataset;
- \( \max(y) \): the maximum value in the dataset.

The autoencoder was compiled using the Adam optimizer [40] and binary cross-entropy loss function. Adam is an optimization algorithm for updating network weights iteratively in response to the training data. Adam provides an optimization solution to noisy problems that handles sparse gradients by combining the Adaptive Gradient Algorithm and Root Mean Square Propagation. The binary cross-entropy loss function was utilized instead of root mean squared error because the last layer of the proposed autoencoder uses a sigmoid activation function. The binary cross-entropy loss function (bce) is calculated according to

\[ bce = -\frac{1}{N} \sum_{i=1}^{N} (y_i \cdot \log(\hat{y}_i) + (1-y_i) \cdot \log(1 - \hat{y}_i)), \]  

where \( N \) is the number of sounds, \( \hat{y}_i \) is the output value of the model and \( y_i \) is the corresponding target value.

### IV. Results

The autoencoder was implemented using Keras and Tensorflow. For the conversion of audio files into readable arrays, the input sampling rate was 50,000 Hz. Each specific category of the dataset was trained with the same autoencoder to denoise the sound. During the decoding process, the autoencoder computes and minimizes the difference between the original noise-free sound (pure sound) and the reconstructed sound (denoised sound).

The mean square error (MSE) was used as the quantitative evaluation criteria for the model. The MSE is the average of the squared difference between the original sounds and denoised sounds generated by the autoencoder and is calculated as follows

\[ MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2, \]  

where \( n \) is the total number of sounds in the test set, \( \hat{y}_i \) is the output value of the model, and \( y_i \) is the corresponding target value. A low MSE indicates more accurate forecasting by the autoencoder.

#### A. Denoising normal function sounds

Seventy percent of the sounds in the normal function category were used for training and validation (34 sounds) and thirty percent for testing (15 sounds). The autoencoder converges to an acceptable validation loss of 0.66 after 6 epochs. There was little improvement in the autoencoder’s performance after 6 epochs. To find out how well the model works, noisy sounds from the test set were fed into the trained autoencoder to denoise.

For random Gaussian noise, the waveforms of the original noise-free sounds, noisy sounds, and denoised sounds are visualized in Fig. 1a. The corresponding STFT power spectrum is visualized in Fig. 1b. The STFT power spectrum of the reconstruction sounds (denoised sounds) from the trained autoencoder is visually much more similar to the STFT power spectrum of the original sounds in the dataset. The MSEs of denoised and original signals in the normal function category are in the range of 0.11 and 0.14, as shown in Fig. 3a.

For water faucet noise, the waveforms of the original noise-free sounds, noisy sounds, and denoised sounds are visualized in Fig. 2a. The corresponding STFT power spectrum is visualized in Fig. 2b. Also in this case the STFT power spectrum of the denoised sound using an autoencoder is quite similar to the original noise-free sound. The MSEs when denoising 15 testing sounds with water sink faucet noise in the normal function category are in the range of 0.09 and 0.14, as shown in Fig. 3b.

#### B. Denoising horizontal misalignment fault sounds

The same proposed autoencoder was trained and tested with the horizontal misalignment fault category. Seventy percent of the horizontal misalignment fault sounds were used for training and validation (137 sounds) and thirty percent for testing (60 sounds). The autoencoder converges to a validation loss of 0.60 after 2 epochs.

For random Gaussian noise, the waveforms of the original noise-free sounds, noisy sounds, and denoised sounds are visualized in Fig. 4a. The corresponding STFT power spectrum is visualized in Fig. 4b. The MSEs of the 60 horizontal misalignment fault sounds in the test set after denoising are in the range of 0.05 and 0.12, as shown in Fig. 6a.

For water faucet noise, the waveforms of the original noise-free sounds, noisy sounds, and denoised sounds are visualized in Fig. 5a. The corresponding STFT power spectrum is visualized in Fig. 5b. The MSEs of the 60 horizontal misalignment fault sounds in the test set after denoising are in the range of 0.09 and 0.15, as shown in Fig. 6b.

### V. Discussion

The low MSE in all cases indicates a high level of similarity between the denoised and original sounds. This means that the noise has been significantly reduced in the signals. The results were similar for all of the sounds in the test set.

When denoising sounds in the normal function category, the MSE on the test set demonstrates that the proposed autoencoder performed better with the water sink faucet noise.
than the random Gaussian noise. The reason for this is that the waveforms of normal sounds are relatively similar, and to train the denoising autoencoder, the same type of water sink faucet noise is added to all normal sounds. As a result, the trained autoencoder can denoise water sink faucet noises with ease. Meanwhile, because random Gaussian noise is generated at random, sounds using the random Gaussian noise may have a variety of waveforms.

In the horizontal misalignment fault category, the suggested autoencoder denoises the random Gaussian noise better than the water sink faucet noise. The reason for this is that there are four different sorts of horizontal misalignment fault sounds in the horizontal misalignment fault category (0.5mm, 1.0mm, 1.5mm, and 2.0mm). As a result, the waveform of sounds in this category differs, making autoencoder training to locate a common pattern amongst these corresponding noisy sounds more difficult.

The water sink faucet noise is clearly audible and overpow-
ers the motor sound when we listen to the noisy sound with the water sink faucet noise. Despite the fact that the autoencoder was able to greatly minimize the flowing water sounds, some remained after the denoising process.

VI. CONCLUSION

In conclusion, this paper presents a promising approach to denoise machine sounds using an autoencoder. The trained model effectively removes noise from the original sounds, with minimal distortion, and maintains the distinguishing features of the sounds. In the scope of this research, only 49 sounds in the normal function category and 197 horizontal misalignment fault sound from the Machinery Fault Database have been used. The proposed technique can be particularly useful in environments where specific background noises are expected, and the model can be trained to denoise those noises specifically. However, it should be noted that the autoencoder might not be able to remove noise perfectly, and there may be trade-offs in the amount of loss tolerable in the reconstructed output.
Therefore, it is crucial to monitor the model’s performance continuously and update hyperparameters as necessary to ensure that the denoising process remains effective. Further investigations are necessary to assess the applicability of the proposed technique to other categories of the Machinery Fault Database and other industrial settings. Nonetheless, this study provides a promising solution to improve the quality of machine sounds for further processing in machine fault diagnosis systems.

REFERENCES

[1] L. Wei, Z. Qian, and H. Zariepour, “Wind turbine pitch system condition monitoring and fault detection based on optimized relevance vector machine regression,” IEEE Transactions on Sustainable Energy, vol. 11, no. 4, pp. 2326–2336, 2020.

[2] R. Hu, J. Wang, A. R. Mills, E. Chong, and Z. Sun, “High-frequency voltage injection based stator interturn fault detection in permanent magnet machines,” IEEE Transactions on Power Electronics, vol. 36, no. 1, pp. 785–794, 2021.

[3] D. Yu, L. Deng, J. Droppo, J. Wu, Y. Gong, and A. Acero, “A minimum-mean-square-error noise reduction algorithm on mel-frequency cepstra for robust speech recognition,” in 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, 2008, pp. 4041–4044.

[4] T. Tosanguan, R. J. Dickinson, and E. M. Drakakis, “Modified spectral subtraction for de-noising heart sounds: Interference suppression via spectral comparison,” in 2008 IEEE Biomedical Circuits and Systems Conference, 2008, pp. 29–32.

[5] K. Ozawa, K. Watanabe, and S. Sakamoto, “Suppression of two noise sources in the same direction by instantaneous spectral subtraction,” in 2019 IEEE 8th Global Conference on Consumer Electronics (GCCE), 2019, pp. 745–746.

[6] K. M. Jeon, N. I. Park, H. K. Kim, M. K. Choi, and K. I. Hwang, “An mdct-domain audio denoising method with a block switching scheme,” IEEE Transactions on Consumer Electronics, vol. 59, no. 4, pp. 818–824, 2013.

[7] M. K. Mathai and J. Deepa, “Design and implementation of restoration techniques for audio denoising applications,” in 2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS), 2015, pp. 21–26.

[8] A. L. M. Levada and D. C. Corrêa, “An adaptive approach for contextual audio denoising using local fisher information,” in 2011 IEEE International Symposium of Circuits and Systems (ISCAS), 2011, pp. 125–128.

[9] X. Yousheng and H. Jianwen, “Speech enhancement based on combination of wiener filter and subspace filter,” in 2014 International Conference on Audio, Language and Image Processing, 2014, pp. 459–462.

[10] Y. Gui and H. Kwan, “Adaptive subband wiener filtering for speech enhancement using critical-band gammatone filterbank,” in 48th Midwest Symposium on Circuits and Systems, 2005., 2005, pp. 732–735 Vol. 1.

[11] G. Yu, S. Mallat, and E. Bacry, “Audio denoising by time-frequency block thresholding,” IEEE Transactions on Signal Processing, vol. 56, no. 5, pp. 1830–1839, 2008.

[12] J. Shi, X. Yu, Y. Wang, W. Wan, and R. Swaminathan, “Noise reduction based on nearest neighbor estimation for audio feature extraction,” in 2012 International Conference on Audio, Language and Image Processing, 2012, pp. 768–771.

[13] G. Baradivish, G. Evangelista, O. Svensson, and F. Sofya, “Pse-svd based audio denoising,” in 2012 5th International Symposium on Communications, Control and Signal Processing, 2012, pp. 1–6.

[14] M. A. Ali and P. M. Shemi, “An improved method of audio denoising based on wavelet transform,” in 2015 International Conference on Power, Instrumentation, Control and Computing (PICC), 2015, pp. 1–6.

[15] “Wavelet signal denoiser.” [Online]. Available: https://www.mathworks.com/help/wavelet/gs/choose-a-wavelet.html

[16] V. Jain and S. Seung, “Natural image denoising with convolutional networks,” in Advances in Neural Information Processing Systems, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds., vol. 21. Curran Associates, Inc., 2009.

[17] Y. Zhao, Z.-Q. Wang, and D. Wang, “Two-stage deep learning for noisy-reverberant speech enhancement,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 27, no. 1, pp. 53–62, 2019.

[18] D. Liu, P. Smaragdis, and M. Kim, “Experiments on deep learning for speech denoising,” in INTERSPEECH, 2014.

[19] S. R. Park and J. Lee, “A fully convolutional neural network for speech enhancement,” in INTERSPEECH, 2017.

[20] Y.-L. Shen, C.-Y. Huang, S.-S. Wang, Y. Tsao, H.-M. Wang, and T.-S. Chi, “Reinforcement learning based speech enhancement for robust speech recognition,” in ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 6750–6754.

[21] R. Fakoor, X. He, I. Tashev, and S. Zarar, “Reinforcement learning to adapt speech enhancement to instantaneous input signal quality,” ArXiv, vol. abs/1711.10791, 2017.

[22] N. Almadari, E. Loburinas, and N. Kehtarnavaz, “Personalization of hearing aid compression by human-in-the-loop deep reinforcement learning,” IEEE Access, vol. 8, pp. 203 503–203 515, 2020.

[23] Y. Wang and D. Wang, “Towards scaling up classification-based speech separation,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 21, no. 7, pp. 1381–1390, 2013.

[24] A. E. Ilesanmi and T. C. Desamrock, Methods for image denoising using convolutional neural network: A review,” Jun 2021. [Online]. Available: https://link.springer.com/article/10.1007/s40747-021-00428-4

[25] M. Tu, Y. Tang, J. Huang, X. He, and B. Zhou, “Towards adversarial learning of speaker-invariant representation for speech emotion recognition,” ArXiv, vol. abs/1903.09606, 2019.

[26] J. M. Valin, “A hybrid dp/depth learning approach to real-time full-band speech enhancement,” in 2018 IEEE 20th International Workshop on Multimedia Signal Processing (Misp), 2018, pp. 1–5.

[27] N. Nie, S. Liang, B. Liu, Y. Zhang, W. Liu, and J. Tao, “Deep Noise Tracking Network: A Hybrid Signal Processing/Deep Learning Approach to Speech Enhancement,” in Proc. Interspeech 2018, 2018, pp. 3219–3223.

[28] L. Fan, F. Zhang, H. Fan, and C. ming Zhang, “Brief review of image denoising techniques,” Visual Computing for Industry, Biomedicine and Art, vol. 2, 2019.

[29] M. I. A. Khalaf and J. Q. Gan, “Deep classifier structures with autoencoder for higher-level feature extraction,” in IJCCI, 2018, pp. 31–38.

[30] J. H. Song, J. H. Kim, and D. H. Lim, “Image restoration using convolutional denoising autoencoder in images,” The Korean Data & Information Science Society, vol. 31, no. 1, pp. 25–40, 2020.

[31] X. Ye, L. Wang, H. Xing, and L. Huang, “Denoising hybrid noises in images with stacked autoencoder,” in 2015 IEEE International Conference on Information and Automation, 2015, pp. 2720–2724.

[32] X. Tian, K. Team, “Keras documentation: Convolutional autoencoder for image denoising.” [Online]. Available: https://keras.io/examples/vision/autoencoder/

[33] H.-T. Chiang, Y.-Y. Hsieh, S.-W. Fu, K.-H. Hung, Y. Tsao, and S.-Y. Chien, “Noise reduction in ecg signals using fully convolutional denoising autoencoders,” IEEE Access, vol. 7, pp. 60 806–60 813, 2019.

[34] P. Xiong, H. Wang, M. Liu, S. Zhou, Z. Hou, and X. Liu, “Ecg signal enhancement based on improved denoising auto-encoder,” Engineering Applications of Artificial Intelligence, vol. 52, pp. 194–202, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197616300331

[35] K. Antczak, “Deep recurrent neural networks for ecg signal denoising,” Jan 2019. [Online]. Available: https://arxiv.org/abs/1807.11551

[36] J. Wang, W. Wang, F. Luo, and S. Wei, “Modulation classification based on denoising autoencoder and convolutional neural network with gnu radio,” The Journal of Engineering, 2019.

[37] E. Almazrouei, G. Gianini, N. I. Almoosa, and E. Damiani, “A deep learning approach to radio signal denoising,” 2019 IEEE Wireless Communications and Networking Conference Workshop (WCNCW), pp. 1–8, 2019.

[38] E. Almazrouei, G. Gianini, C. Mio, N. I. Almoosa, and E. Damiani, “Using autoencoders for radio signal denoising,” Proceedings of the 15th ACM International Symposium on QoS and Security for Wireless and Mobile Networks, 2019.

[39] M. Faisalda: Machinery fault database. [Online]. Available: http://www02.smt.ufrj.br/~joffshore/mds/page_01.html

[40] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2017.