Leveraging end-to-end denoisers for denoising periodic signals

Jules Rio, Olivier Alata, Fabien Momey, Christophe Ducottet
Laboratoire Hubert Curien, CNRS UMR 5516, Université Jean Monnet, IOGS
Université de Lyon
42023 Saint-Étienne, France

***Abstract*—In this paper, we propose a new framework for denoising 1D periodic signals with deep learning models by exploiting their periodic properties. Our method lies on a transformation of the raw waveform into a grid containing the different periods. Networks used with these data can be simply obtained by leveraging end-to-end fully convolutional denoisers containing only 1D convolutions, by replacing some of their layers by 2D convolutions. Our method also offers the advantage of being able to learn one model for generalizing to a large band of frequencies, including unseen ones, instead of requiring to learn one model per frequency. We also study the generalization of our method to real data.**

***Index Terms*—Denoising, periodic signals, deep learning, white noise, colored noise**

I. Introduction

The denoising task, which consists of extracting a clean signal $\mathbf{y}$ from a mixture $\mathbf{x} = H(\mathbf{y}) + \mathbf{e}$ with $\mathbf{e}$ a noise and H the transfer function of the sensor (in this paper, we will consider $H(\mathbf{y}) = \mathbf{y}$), has been widely explored in the last decades. It is often a prerequisite to other tasks such as fault diagnosis [1], [2] or automatic speech recognition [3].

Recently, conventional methods [4] have widely been replaced by deep learning techniques, especially in the context of audio signals. Several architectures have been tried, including convolutional networks [5]–[10], generative adversarial networks [11]–[14] or LSTM [3]. Audio source separation, which can be seen as a generalization of audio enhancement has also benefited from Deep Learning methods [15]–[19].

In this paper, we focus on denoising periodic and more generally cyclostationary signals. Periodicity can be encountered in various signals, e.g. audio signals (where the periodicity might be restrained to a few seconds), electrocardiograms and rotating machinery (where the periodicity might be expected for the whole signal). For exploiting the periodicity, one solution is to use the coefficients of the Fourier series [20]. Time-frequency representations [21] are also commonly used, however they only give information about the local spectral content. Concerning neural networks (especially convolutional ones), they generally focus mostly on an area around the observed sample due to their limited receptive field. Observing several periods could be done using dilation or long convolutions, however, this would allow specializing the network on denoising periodic signals with a certain range of frequencies, but it would not be possible to obtain a single network able to generalize to a large band of frequencies.

In this paper, our contribution is to propose a new framework for improving 1D convolutional denoisers when applied to periodic data, by reshaping the raw data to form a 2D grid where each line is a realisation of the same signal at the same timestamp. This method can be applied to any 1D convolutional denoisers and allow to train a single model for analyzing signals of various frequencies. To exploit this new shape, we propose some adaptations to a given deep learning model designed for end-to-end denoising [7], [15] in order to observe not only the temporal dimension, but also the newly created periodic dimension.

In Section II, we detail our method and the way of adapting neural networks to the new shape of our data. In Section III, we give an example of adapting a 1D convolutional denoiser for our method and we show the improvements that are possible in an ideal case, where the signals are exactly periodic and the fundamental frequency is known. In Section IV, we show how our method could deal with more realistic setups.

II. Exploiting the periodicity of the data

A. Introduction of the general framework

We aim to extract a component $\mathbf{y}$ of period T from the mixture $\mathbf{x} = \mathbf{y} + \mathbf{e}$ where $\mathbf{e}$ is a noise. If we assume that first and second order moments of $\mathbf{e}$ are periodic (that is to say that $\mathbf{e}$ is cyclostationary), then the sequence $\{x(t + kT)\}_{k\in\mathbb{Z}}$ (with a fixed t) is the realisation of a stationary (at the second order) process. Additionally, if $\mathbf{e}$ is centered, we can expect the periodic-wise sequence $\{e(t + kT)\}_{k\in\mathbb{Z}}$ to be close to the realisation of a white noise as correlations of the noise tend to decrease with time. Therefore, using the information contained in this periodic-wise sequence additionally to the one contained in a temporal sequence of samples around $x(t)$ should be beneficial in terms of denoising results. For neural networks to be able to benefit from this information, we reshape the data to form a grid $\mathbf{X} = \{X_{p,n}\}_{p\in[\![1,P]\!],n\in[\![1,N]\!]}$ where P is the amount of realisations of N samples and $X_{p,n} = x\left(nT_e + t_0 + (p-1)T\right)$ with T_e the sampling period and t_0 an initial time. By doing this, we ensure that for two given p_1 and p_2, $X_{p_1,n}$ and $X_{p_2,n}$ always share the same probabilistic properties. Notice that there is no necessity for N to be the amount of samples in one period of the signal, which

This work was supported by the IMOTEP project within the 'Investissements d'Avenir' program operated by the French Environment and Energy Management Agency (ADEME)

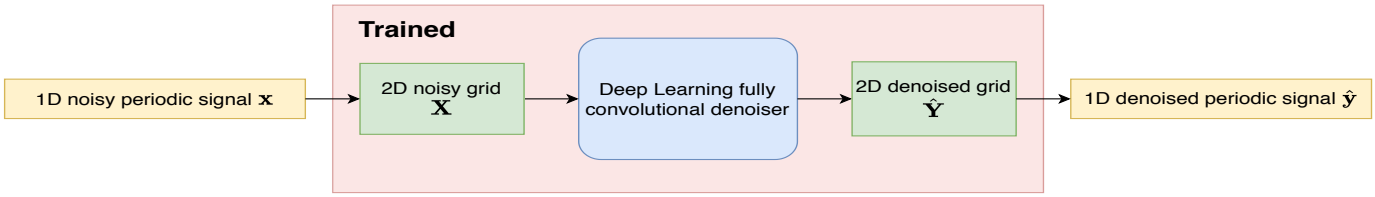


Fig. 1. Global framework for denoising

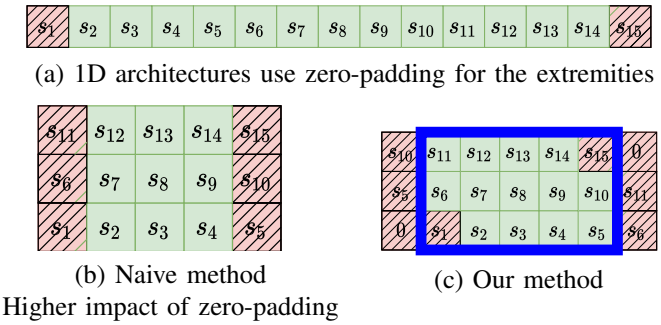
means we can generate grids of the same size for different frequencies or have more than one single complete period in a single line (we show how these properties are beneficial in II-B and II-C).

For exploiting this new shape of the data, neural networks also need to be able to consider not only the temporal dimension, but also the newly created periodic dimension. A possibility would be to use a neural network used for 2D denoising. However, such neural networks generally treat the two dimensions the same way, whereas we can expect our method to require a longer receptive field on the temporal dimension than on the periodic dimension to be able to recognize some specific shapes. Additionally, in real applications, the amount of periods might be limited and the properties of the signal could evolve sharing only few common information, which means it is preferable to avoid having a too long receptive field on the periodic dimension. The method we adopt is to take a 1D convolutional neural network designed for end-to-end denoising [5]–[8] and to replace some of its 1D convolutions by 2D convolutions. Here again, we want to avoid a too long receptive field on the periodic dimension, which means we prefer to have a limited amount of 2D convolutions.

The resulting framework for performing the denoising is shown in Figure 1. This method could be compared to patch-based denoising, a method often used for image denoising, including Deep Learning methods [22], where our patches are obtained directly through assumptions on periodicity.

B. Shaping data for training

As explained in II-A, it is not necessary that the length of the line coincides with one period of the original signal, as long as the process obtained by concatenating the resulting lines remains periodic (for example, the process obtained by removing the second half of each period remains periodic). This implies that we can create grids of the same size for signals of different frequencies as long as we ensure that all lines of the grid share the same phase, allowing us to train a single model for generalizing to a large band of frequencies. This process is applicable with real data or with synthetic data where each line is generated as the sum of a deterministic signal and a random noise. Notice that due to the decorrelation along the periodic dimension, using synthetic data with a white noise could lead to good generalization for centered colored noises (a demonstration of this will be given in Section III-C).



(a) 1D architectures use zero-padding for the extremities

(b) Naive method
Higher impact of zero-padding

(c) Our method

Fig. 2. Method for creating the grid with a signal of 15 samples containing 3 periods of 5 samples, with networks having a receptive field of length 3 for the first dimension. Samples affected by zero-padding on this dimension are hatched.

C. Shaping the data for inference

For denoising the signals, the first step is to transform the 1D signal into a grid with data from different periods being in parallel. If the signal is made of P periods, each one having N samples, the most natural method would be to create P lines, each one containing exactly one period. An example of the resulting grid is given in Figure 2(b) for the signal of Figure 2(a). However, as it is shown in those figures, this would result in using more zero-padding along the temporal dimension than in the 1D method (zero-padding would be performed for each line instead of only at each extremity). Therefore, we propose (see Figure 2(c)) to extend each line by the length of the temporal receptive field on each side of the grid, by adding the end of the previous period at the beginning of the line and the beginning of the next period at the end. Only the first and last lines would have to be filled with zeros. The pass through the network would then require additional zero-padding, but impact would be limited as only the center of the denoised grid is further used for forming the resulting 1D signal. In Figure 2(c), only the sub-grid inside the bold blue box will be used to create the 1D signal, resulting in the same amount of zero-padding for the useful part as in the 1D process.

Here, we did not mention zero-padding regarding the periodic dimension. However, the problem would also be present, and our method requires a sufficient amount of periods to give satisfying results. Also notice that in this paper, we assume previously knowing the exact frequency of the signals, which means no period detection and registration are required.

III. Experiments

A. Settings and training

For comparing our method with a 1D end-to-end neural network, we will use a setup similar to the one of [23]. We train our models with data generated through the model :

$$s_n = \lambda \cos(2\pi\nu n + \phi_0) + (1-\lambda) sgn(\cos(2\pi\nu n + \phi_1)) \quad (1)$$

where $\lambda \in [0,1]$, $(\phi_0, \phi_1) \in [0, 2\pi]^2$ and $\nu = \frac{f}{f_e}$ with f the frequency and f_e the sampling frequency (10^5Hz in our studies).

This model is used for testing the ability of a network trained with a specific shape of signals to adapt to other kind of signals and was originally used in [23] to consider both smoothness and strong discontinuities.

The noisy signal is created by adding a noise $\mathbf{e}$ to the signal with a power defined to obtain a pre-defined signal-to-noise ratio (SNR), defined as:

$$SNR = 10 \log_{10} \left(\frac{||\mathbf{s}||_2^2}{||\mathbf{e}||_2^2} \right) \quad (2)$$

The neural networks are based on [7]. We consider two models from [23] : the "full" architecture, that we will denote as "Original", and the "medium" architecture trained with a regularization parameter of 1, that we will denote as "Medium". We will create two 2D architectures based on the same architecture. Our first model is the same as the "Medium" one, where we replace the 1D convolutions of layers 4 and 8 of each stack by 3×3 convolutions (without dilation for the periodic dimension). We will denote this architecture as PWNet for Periodic WaveNet as we used the architecture of WaveNet (another architecture could be used). As using 2D kernels for some layers will expand the amount of parameters and might give an unfair comparison with the "Medium" architecture, we also create a smaller architecture based on the architecture denoted as "small" in [23] by replacing the layers 3 and 6 of each stack by 2D convolutions. We will denote this architecture as PWNet^s. Both PWNet and PWNet^s have a receptive field of 9 periods for the periodic dimension.

1D networks are trained the same way as described in the original paper [23]. Concerning our 2D architectures, we use the following settings with grids of synthetic signals generated with Equation 1. We use 10000 grids, containing $P = 10$ realisations of $N = 2048$ samples with frequencies in set $\{0.25, 0.5, 1, 2, 4, 8, 16, 24, 32\}$Hz and a white noise for reaching SNR in set $\{-10, -5, 0, 5, 10, 20\}$dB. We perform 30 epochs, with batches of size 8, an initial learning rate of 0.001 divided by 10 after 20 epochs. 3000 validation grids (generated as the training ones) are used to detect the best model. The sampling frequency of the signals is $f_e = 10^5$Hz.

Most of these settings are very close to the ones of [23] and the total amount of samples in each grid (20480) is in the same order as the amount of samples in each signal used for training in [23] (20000).

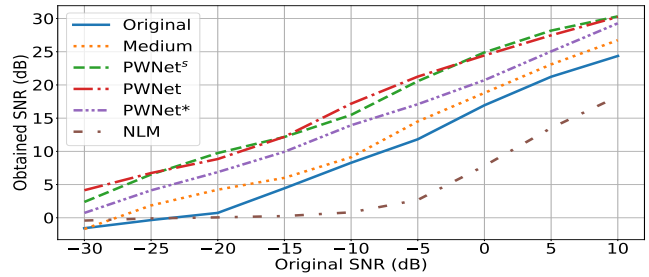


Fig. 3. Evolution of the obtained SNR for various initial SNR, with signals of fundamental frequency of 60Hz

Our loss term for 2D models is the mean absolute error (MAE), defined in our case as :

$$\mathcal{L}(\hat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{P \times N} \sum_{p=1}^{P} \sum_{n=1}^{N} |\hat{Y}_{p,n} - Y_{p,n}| \quad (3)$$

with $\hat{Y}$ the predicted grid and Y the ground truth.

B. Results with synthetic friction data

As explained previously, we want to check the ability of trained models to adapt to different kind of signals. Therefore, we apply our models to data generated with the following friction model from [24]:

$$\mu(V) = \left(\mu_c + (\mu_s - \mu_c) e^{-\left(\frac{|V|}{v_s}\right)} \right) \times sgn(V) + k_s V \quad (4)$$

where μ_c, μ_s, i and k_s are constants and V is a periodic speed (we generate it with Equation 1 and multiply it by a coefficient proportional to the frequency such that the speed's maximal value is 1 at 40Hz). As for the training data, this model will generate both discontinuities and smooth variations, but the variations will be different from a sinusoid.

Each evaluation is performed on 60 signals containing white noise, all signals having the same SNR and the same frequency. All generated signals contain 100000 samples at sampling rate 10^5Hz and the grids are created based on the theoretical frequency. To compare our method with a standard non-deep learning based method, We also apply the method of [25] with a bandwidth $\lambda = 0.6\sigma$ (with σ the standard deviation of the noise), a patch half-width of 500 samples and a neighborhood half-width of 10000 samples (which allows exploiting the periodicity for high frequencies). We denote this method as "NLM". There might be more efficient methods, but this one is well suited for denoising periodic signals containing white noise.

Figure 3 (PWNet* will be introduced in Section IV) shows the evolution of the obtained SNR depending on the initial SNR with a fundamental frequency of 60Hz. 2D models largely outperform 1D models, as well as [25].

Figure 4 shows the evolution of the SNR with respect to the frequency of the signal for an initial SNR of -5dB. For very low frequencies, our method is outperformed by 1D models (including [25]), which could be expected as the amount of periods is low and 2D models cannot use the periodic

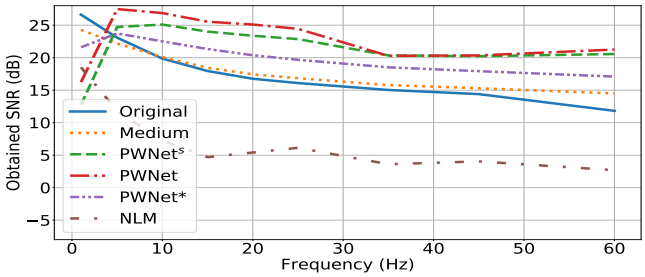


Fig. 4. Evolution of the obtained SNR for various fundamental frequencies, with original SNR of -5dB

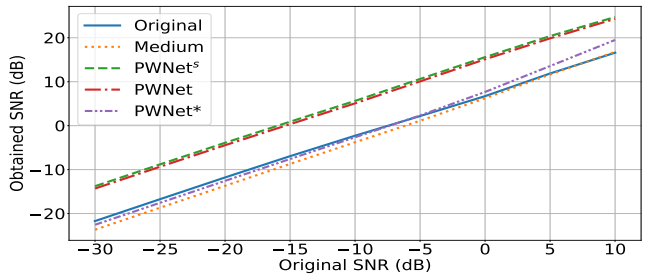


Fig. 5. Evolution of the obtained SNR for various initial SNR, with signals of fundamental frequency of 60Hz and colored noise

dimension properly. 2D models become advantageous when the amount of periods is large enough to use the periodic dimension. The better results of PWNet compared to PWNet^s might be due to the higher depth, which implies a longer receptive field along the temporal dimension. However, the difference remains small and the comparison with 1D models indicates that the periodic dimension is more useful for this experiment.

C. Generalization to colored noises

To confirm the assumption of a better generalization to colored noises due to whitening the noise along the periodic dimension, we perform a new set of evaluations, where the only difference with Section III-B is that we do not use a white noise but a noise generated according to:

$$\mathbf{e} = \gamma \mathbf{c} + (1-\gamma)\mathbf{w} \tag{5}$$

where $\mathbf{w}$ is a white noise and $\mathbf{c}$ is a colored noise obtained by performing a moving average with a square window of 200 samples on a white noise, which could be assimilated to filtering a white noise with a low-pass filter with cutoff frequency 221.5Hz (with a sampling frequency of 10^5Hz). We use $\gamma = 0.9$ to get largely colored noises.

The evolution of the obtained SNR for various initial SNR with a fundamental frequency of 60Hz is shown on Figure 5. We also give some numerical values in Table I.

As expected, our method improves generalization to this kind of noise. An illustration of denoising results of such signals with both 1D and 2D methods is shown in Figure 6, illustrating phenomenons explaining these quantitative results.

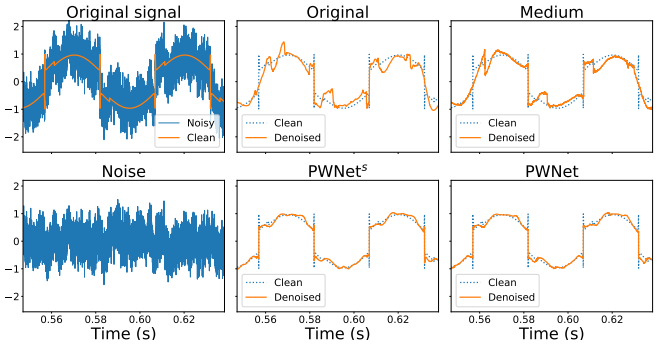


Fig. 6. Example of denoising for a signal containing colored noises

TABLE I
NUMERICAL VALUES OF SNR (IN DB) FOR SIGNALS AT 60HZ WITH COLORED NOISE

Initial SNR	Medium	Original	PWNet	PWNet^s
-20	-13.68	-11.78	-4.44	**-3.89**
-5	1.12	2.19	10.07	**10.63**
5	11.65	11.88	19.87	**20.39**

Notice that in this cas, PWNet^s gives a better generalization to colored noises.

IV. GENERALIZATION TO REAL SIGNALS

Based on the results showed in Section III, our method seems advantageous. However, as the training signals are periodic and the fundamental frequency is known, it is likely that these models will not be as efficient when dealing with real data. To illustrate this problem, we applied PWNet and Medium to real friction signals obtained by Ireis (HEF Group)[1] with a linear reciprocating tribometer, for which the frequency used to create the grids was the one given as a reference to the mechanism. To show how our method remains effective, we also trained a new 2D model, that we also applied to the real data. The only difference of this architecture compared to the PWNet lies in the training set, as the lines of the grids will contain a constant shift compared to the theoretical value, defined by a uniform law $\mathcal{U}([-0.05S_T; 0.05S_T])$ with S_T the amount of samples per period. We denote this new architecture as PWNet*.

Figure 7 shows the results obtained with a signal containing almost no noise. Despite the weak noise, assuming exactly periodic signals results in poor performance near the peaks. PWNet* manages to get similar results to the ones of the 1D method, showing that it is able to deal with periodic shifts by focusing on the local context when the periodicity is not exploitable.

Figures 3 and 4 seem to indicate that PWNet* remains able to benefit at least partially from the periodicity, as it gets better results as 1D models for these exactly periodic signals. With colored noises (Figure 5), PWNet* does not seem to be able to exploit the periodicity and when the initial SNR is low, but it still performs similarly to 1D models. When the initial

[1] http://www.ireis.fr/en/

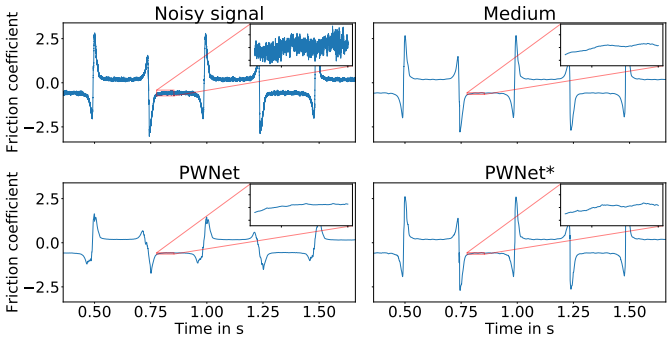


Fig. 7. Comparison of the methods with a real signal containing weak noise

TABLE II
RESULTS (IN DB) WITH AN ERROR IN THE GIVEN FREQUENCY

Frequency	**20Hz**		**45Hz**		**60Hz**	
Initial SNR	**-10**	**10**	**-10**	**10**	**-10**	**10**
Medium	12.38	31.56	10.00	28.96	9.11	26.74
PWNet	14.44	17.21	**14.43**	20.15	**14.37**	20.93
PWNet*	**14.88**	**32.35**	14.23	**30.82**	13.69	**29.11**

SNR gets higher, PWNet* starts exploiting the periodicity and outperforms 1D models.

Additionally, we evaluate Medium, PWNet and PWNet* with the same signals as for Section III-B, but instead of creating the grid with the exact fundamental frequency, we create it with a frequency that has up to 1% relative difference with the correct frequency, which will result in creating grids with shifts between the lines. We show some values in Table II. With an initial SNR of -10dB, PWNet and PWNet* perform similarly and largely outperform Medium, showing that in the presence of large noise, 2D models remain interesting even if they were trained with exactly periodic signals and by knowing the exact frequency. When the initial SNR grows, PWNet seems to saturate and is outperformed by both PWNet* and Medium, which shows that the assumption of exactly periodic signals degrades the results in case of weak noise. PWNet* still outperforms Medium, showing that this model is still able to benefit from periodicity even if the phenomenons are not exactly at the same position in the lines.

V. CONCLUSION

In this paper, we proposed a method for denoising signals whose useful information is assumed to be periodic. We demonstrated the efficiency of our method compared to end-to-end denoisers in some tool applications, where ground truth was exactly periodic. We also showed that with simple changes of the training data, our method might be able to benefit from the periodic information even for real data where the lines will contain some shifts (due to a bad detection of the fundamental frequency, local events, ...). Further work will focus on improving this generalization to realistic contexts.

REFERENCES

[1] X. Wang, Y. Zi, and Z. He, "Multiwavelet denoising with improved neighboring coefficients for application on rolling bearing fault diagnosis," *MSSP*, vol. 25, no. 1, pp. 285–304, 2011.

[2] S. Abbasion, A. Rafsanjani, A. Farshidianfar, and N. Irani, "Rolling element bearings multi-fault classification based on the wavelet denoising and support vector machine," *MSSP*, vol. 21, no. 7, pp. 2933–2945, 2007.

[3] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J.R. Hershey, and B. Schuller, "Speech enhancement with lstm recurrent neural networks and its application to noise-robust asr," in *LVA/ICA*. Springer, 2015, pp. 91–99.

[4] J. Chen, J. Benesty, Y.A. Huang, and E.J. Diethorn, "Fundamentals of noise reduction," in *Springer Handbook of Speech Processing*, pp. 843–872. Springer, 2008.

[5] S.W. Fu, T.W. Wang, Y. Tsao, X. Lu, and H. Kawai, "End-to-end waveform utterance enhancement for direct evaluation metrics optimization by fully convolutional neural networks," *IEEE/ACM TASLP*, vol. 26, no. 9, pp. 1570–1584, 2018.

[6] S. Fu, Y. Tsao, X. Lu, and H. Kawai, "Raw waveform-based speech enhancement by fully convolutional networks," in *2017 APSIPA ASC*. IEEE, 2017, pp. 006–012.

[7] D. Rethage, J. Pons, and X. Serra, "A wavenet for speech denoising," in *2018 ICASSP*. IEEE, 2018, pp. 5069–5073.

[8] L. Casas, A. Klimmek, N. Navab, and V. Belagiannis, "Adversarial signal denoising with encoder-decoder networks," *arXiv preprint arXiv:1812.08555*, 2018.

[9] A. Neacsu, K. Gupta, J.P. Pesquet, and C. Burileanu, "Signal denoising using a new class of robust neural networks," in *2020 EUSIPCO*. IEEE, 2021, pp. 1492–1496.

[10] Ritwik Giri, Umut Isik, and Arvindh Krishnaswamy, "Attention wave-u-net for speech enhancement," in *2019 WASPAA*. IEEE, 2019, pp. 249–253.

[11] C. Donahue, B. Li, and R. Prabhavalkar, "Exploring speech enhancement with generative adversarial networks for robust speech recognition," in *2018 ICASSP*. IEEE, 2018, pp. 5024–5028.

[12] S. Pascual, A. Bonafonte, and J. Serra, "SEGAN: Speech enhancement generative adversarial network," *arXiv preprint arXiv:1703.09452*, 2017.

[13] Z. Meng, J. Li, Y. Gong, and B.H.F. Juang, "Cycle-consistent speech enhancement," *arXiv preprint arXiv:1809.02253*, 2018.

[14] S. Abdulatif, K. Armanious, K. Guirguis, J. T Sajeev, and B. Yang, "Aegan: Time-frequency speech denoising via generative adversarial networks," in *2020 EUSIPCO*. IEEE, 2021, pp. 451–455.

[15] D. Stoller, S. Ewert, and S. Dixon, "Wave-u-net: A multi-scale neural network for end-to-end audio source separation," *arXiv preprint arXiv:1806.03185*, 2018.

[16] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM TASLP*, vol. 26, no. 10, pp. 1702–1726, 2018.

[17] Z. Rafii, A. Liutkus, F.R. Stöter, S.I. Mimilakis, D. FitzGerald, and B. Pardo, "An overview of lead and accompaniment separation in music," *IEEE/ACM TASLP*, vol. 26, no. 8, pp. 1307–1335, 2018.

[18] Yi Luo and Nima Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM TASLP*, vol. 27, no. 8, pp. 1256–1266, 2019.

[19] Yi Luo, Zhuo Chen, and Takuya Yoshioka, "Dual-path rnn: efficient long sequence modeling for time-domain single-channel speech separation," in *2020 ICASSP*. IEEE, 2020, pp. 46–50.

[20] O. Dromer, O. Alata, and O. Bernard, "Impedance cardiography filtering using scale fourier linear combiner based on rls algorithm," in *2009 Annual International Conference of the IEEE EMBS*. IEEE, 2009, pp. 6930–6933.

[21] J. Gao, R. Wang, L. Hu, and R. Zhang, "A novel manifold learning denoising method on bearing vibration signals," *Journal of Vibroengineering*, vol. 18, no. 1, pp. 175–189, 2016.

[22] I. Hong, Y. Hwang, and D. Kim, "Efficient deep learning of image denoising using patch complexity local divide and deep conquer," *Pattern Recognition*, vol. 96, pp. 106945, 2019.

[23] J. Rio, F. Momey, C. Ducottet, and O. Alata, "Wavenet based architectures for denoising periodic discontinuous signals and application to friction signals," in *2020 EUSIPCO*. IEEE, 2021, pp. 1580–1584.

[24] S. Andersson, A. Söderberg, and S. Björklund, "Friction models for sliding dry, boundary and mixed lubricated contacts," *Tribology international*, vol. 40, no. 4, pp. 580–587, 2007.

[25] Brian H Tracey and Eric L Miller, "Nonlocal means denoising of ecg signals," *IEEE transactions on biomedical engineering*, vol. 59, no. 9, pp. 2383–2386, 2012.