25·29 August 2024
inter.noise
NANTES FRANCE

Trainable signal encoders that are robust against noise

Peter Balazs[1]
Acoustics Research Institute, Austrian Academy of Sciences
Dominikanerbastei 16, 1010 Vienna, Austria

Daniel Haider[2]
Acoustics Research Institute, Austrian Academy of Sciences
Dominikanerbastei 16, 1010 Vienna, Austria

Vincent Lostanlen[3]
Nantes Université, École Centrale Nantes, CNRS, LS2N
UMR 6004, F-44000 Nantes, France

Felix Perfler[4]
Acoustics Research Institute, Austrian Academy of Sciences
Dominikanerbastei 16, 1010 Vienna, Austria

**Alphabetical order, all authors have contributed equally.*

ABSTRACT

Within the deep learning paradigm, finite impulse response (FIR) filters are often used to encode audio signals, yielding flexible and adaptive feature representations. We show that a stabilization of FIR filterbanks with fixed filter lengths (convolutional layers with 1-D filters)leads to encoders that are optimally robust against noise and can be inverted with perfect reconstruction by their transposes. To maintain their flexibility as regular neural network layers, we implement the stabilization via a computationally efficient regularizing term in the objective function of the learning problem. In this way, the encoder keeps its expressive power and is optimally stable and noise-robust throughout the whole learning procedure. We show in a denoising task where noise is present in the input and in the encoder representation, that the proposed stabilization of the trainable filterbank encoder is decisive for increasing the signal-to-noise ratio of the denoised signals significantly compared to a model with a naively trained encoder.

1. INTRODUCTION

In traditional machine learning, the procedure of feature extraction was often considered conceptually separated from the actual learning problem and made use of classical signal

[1]peter.balazs@oeaw.ac.at

[2]daniel.haider@oeaw.ac.at

[3]vincent.lostanlen@ls2n.fr

[4]felix.perfler@oeaw.ac.at

processing methods such as time–frequency transforms, principal component analysis, etc. [1, 2] Nowadays, there is a strong tendency towards designing end-to-end neural network architectures, in which feature extraction becomes an implicit procedure that is seamlessly embedded into the *one* model that aims to solve the whole learning problem at once [3, 4]. In many neural network architectures, this procedure is implemented as an encoder layer that maps the input signals to a suitable feature space and is learned together with the rest of the model parameters [5].

For audio applications, encoders are usually based on the operation of convolution. A standard architecture is a convolutional layer with 1-D filters that maps audio signals in their "raw waveforms" into the encoder space. From a classical signal processing perspective, this map corresponds to a finite impulse response (FIR) filterbank Φ with filters $\phi_j \in \mathbb{R}^N$, $j = 1, \ldots, J$ that decomposes a signal $x \in \mathbb{R}^N$ into the $J \times N$ array given by

$$(\Phi x)[j, n] = (x * \phi_j)[n]. \tag{1}$$

All standard linear time–frequency representations, such as the short-time Fourier transform or the constant-Q transform can be implemented in this way. If the encoder is trainable, the entries of the single filters $\phi_j[k]$ are parameters of the model and are updated via gradient descent so that a given objective function is minimized. Trainable filterbank encodings in end-to-end models have been used successfully for source separation [6] and denoising. However, drawbacks come with this kind of signal encoding, such as less interpretability and controllability of the features that the rest of the model uses to solve the learning problem. The fundamental assumption in this work is that

if noise is involved in the learning procedure it is particularly preferable to have stable and redundant encoder representations.

Such stable and redundant encoders are defined in a frame-theoretical context and are called *tight*. In many signal processing applications, tightness has been shown to be the key property when signal representations are exposed to noise [7–9]. The stability property (condition number close to one) guarantees that small perturbations of the input of the encoder result in small errors in the output, while redundancy acts as a compensation mechanism that allows a better reconstruction from noisy coefficients. In the machine learning context, this can be seen as an optimal setting against adversarial examples since any manipulation of an input that changes the encoding significantly needs to come with a significant change of its magnitude too. A previous publication has shown for an image classification task that this is not only an abstract conceptual advantage [10].

A standard problem where we assume that our assumption applies is when unwanted noise should be removed from a signal, i.e., in a denoising task. This is often done via a masking process in the encoder domain (encoder–mask–decoder model). Classical (non-deep-learning-based) methods are Wiener filtering [11], CASA [12, 13], or block-thresholding methods [14], where the encoder is a time–frequency transform and the decoder its inverse. In a deep learning paradigm, encoder, decoder, and mask are neural networks that are optimized via gradient descent to minimize an objective that reflects the quality of the denoised signal [6, 15].

Noise, however, does not only appear on the input level but also within a neural network. For example, it is known that adding noise to the layers during training can improve the generalization of a model [16]. Also here, we conjecture that tight encoding have advantages. The goal of this work is to identify where these advantages reside.

In Section 2, we discuss the theoretical background of tightness, and in Section 3 we derive a stabilization mechanism that keeps a trainable filterbank encoder tight throughout training. In Section 4, we demonstrate the effectiveness of this mechanism on an encoder–mask–decoder model optimized to denoise audio signals. In an extended setting, additional noise is introduced after the encoder. We can show that a model with a tight encoder reaches a significantly higher

signal-to-noise ratio of the denoised signals to the residual noise than the same model without stabilization.

2. TIGHT FILTERBANK ENCODINGS

We study the robustness of a filterbank encoding Φ against noise via the notion of stability that is associated with the definition of a *frame*. A filterbank Φ is called a frame for $\mathbb{R}^N$ if there are constants $0 < A \leq B$ such that the Lipschitz-type inequality

$$A \cdot \|x\|^2 \leq \|\Phi x\|^2 \leq B \cdot \|x\|^2 \tag{2}$$

holds for all $x \in \mathbb{R}^N$. The optimal upper bound is given by the squared operator norm of Φ and the optimal lower bound by the inverse of the squared operator norm of the pseudoinverse $\Phi^\dagger = (\Phi^\top \Phi)^{-1}\Phi^\top$. Here, $\Phi^\top$ denotes the transposed filterbank of Φ. The condition number associated with Φ is given by $\kappa = B/A$, where A, B are the optimal bounds, and determines the numerical stability of Φ. If $A = B$ then Φ is optimally stable in this sense and is called a *tight frame*. One can think of tight frames as orthonormal bases that allow redundancy, i.e., we have optimal stability for the non-unique encoder coefficients of any input signal.

The central property of a tight frame is that for any $x \in \mathbb{R}^N$ we have the Parseval-like decomposition

$$x = A^{-1}\Phi^\top \Phi x. \tag{3}$$

Therefore, tight frames with $A = 1$ are also known as *Parseval frames*. Note that Equation 3 is equivalent to $\Phi^\dagger = A^{-1}\Phi^\top$, which yields a particularly convenient way to reconstruct any x from its encoding. Reconstruction via $\Phi^\dagger$ is particularly desirable as it provides the reconstruction with minimal Euclidean norm.

Tight frames are known to have particular benefits in the presence of noise [8, 9]. In the context of a filterbank encoder in an end-to-end regime, we discuss two different settings.

Noise in the Encoder. In classical signal processing, it is often assumed that the transmission of encoded signals is tainted with noise. In our context, this kind of perturbation appears in a specific training technique that involves the introduction of artificial noise in a neural network layer to increase the generalizability of the model. For the reconstruction of the input signal from the non-unique noisy coefficients, redundancy has been shown to bring conceptual advantages [7]. In particular, it is known that an encoding by Φ is maximally robust against noise if Φ is a tight frame with unit norm elements, in the following sense:

Theorem 1 (Fickus et al., [9])**.** *Let Φ form a uniform frame, i.e. all elements have equal norm, E be encoder noise that consists of independent draws from a uniform or normal distribution, and let $\tilde{x} = \Phi^\dagger((\Phi x) + E)$ denote the reconstruction from the noisy encoding. The mean-squared error $\mathbb{E}\left[\|x - \tilde{x}\|^2\right]$ is minimal if and only if Φ is tight.*

Alternatively, we can see this by considering the (linear) signal-to-noise ratio $\mathrm{SNR}^*(\Phi x, E) = \|\Phi x\|/\|E\|$ and observing that the reconstruction of the noise source is bounded as

$$\|\Phi^\dagger E\|^2 \leq \frac{\kappa}{\mathrm{SNR}^*(\Phi x, E)}\|x\|^2. \tag{4}$$

Clearly, the RHS in Equation 4 is minimal if $\kappa = A/B = 1$, i.e., Φ is tight.

Noise in the Input. For many applications, clean data is not available. Instead, we have a dataset of noisy signals of the form $x + e$, where x is the ideally clean signal and e an unknown source of noise that should be removed. If Φ is tight we can make use of the energy preservation property to see that

$$\|\Phi(x+e)\|^2 = A \cdot \|x+e\|^2 = A \cdot \left(\|x\|^2 + 2 \cdot \langle x, e\rangle + \|e\|^2\right) \leq A \cdot \left(\|x\|^2 + 2 \cdot \|x\| \cdot \|e\| + \|e\|^2\right).$$

Hence, we can ensure that the noise only affects the energy of the encoder proportional to its own energy. In other words, we have direct control over the energy deviation of the encoding in the sense that input perturbations are always scaled by the same constant A in the encoding. This is particularly interesting in the context of adversarial examples, which are designed in a way that small perturbations have an essential impact on the output of a trained model. In fact, ensuring tightness of all convolutional layers in a neural network with $A = 1$ increases the robustness of the network against adversarial examples in classification tasks [10].

To relate the robustness property from Theorem 1 to this setting we use an encoder–mask–decoder model for the task of denoising. Such a model is closely related to Gabor filters [17], and is known as frame multiplier in mathematics [18]. Formally, it can be written as

$$\mathcal{M}_{\Phi,M,\Psi}(x+e) = \Psi(M \odot (\Phi(x+e))),$$

where $\odot$ denotes the point-wise multiplication, M is a mask for the encoded signals, and Ψ a decoder. In the end-to-end regime, the learning problem consists in optimizing Φ, Ψ, and M such that

$$\mathcal{M}_{\Phi,M,\Psi}(x+e) \approx x$$

for any x "of interest". Let us now assume that we have an oracle that gives us an adaptive mask M_Φ such that

$$M_\Phi \odot \Phi(x+e) = \Phi\tilde{x} + \delta, \tag{5}$$

where the residual noise δ is much smaller than the original noise e, i.e., $\|\delta\| \ll \|e\|$, and the coefficients of the clean signal are not affected too much by the masking, i.e., $\|\Phi\tilde{x}\| \approx \|\Phi x\|$. Then, by Theorem 1, we obtain the best reconstruction of x in terms of $\tilde{x}$ if Φ is a tight frame with unit norm elements and $\Psi = A^{-1}\Phi^\top$. We emphasize that this oracle assumption might be naive and does surely not hold in general. Still, we believe it is reasonable to assume that tightness is beneficial.

Given the theoretical motivation for tightness, we are left with two practical problems when aiming to employ tightness in a deep learning regime:

1) *Unit norm tight frames are very restrictive and inherently difficult to construct deterministically.*

2) *If we start with a tight filterbank, how can we ensure that it stays tight during training?*

We aim to solve these two problems by proposing a stabilization scheme that is tailored to the optimization of trainable filterbank encoders via gradient descent.

3. TRAINABLE TIGHT FILTERBANK ENCODINGS

3.1. Tight Initialization

The unit norm assumption is a strong restriction that is not invariant under small perturbations. Hence, it seems that it is too rigid to be included in the iterative procedure of gradient descent. Therefore, to approach Problem 1) we drop the unit norm assumption and use a standard way of constructing the nearest tight filterbank with $A = 1$ to a given one.

Lemma 1 (Christensen [7])**.** *Let the filterbank Φ with filters ϕ_j be a frame. The filterbank $\Phi^\sharp$ with filters $(\Phi^\top\Phi)^{-\frac{1}{2}}\phi_j$ is the Parseval frame closest to Φ in the Euclidean norm.*

This provides a method to tighten any filterbank frame, i.e., is suitable for any initialization scheme of a trainable encoder as long as it yields a frame. A random initialization is valid.

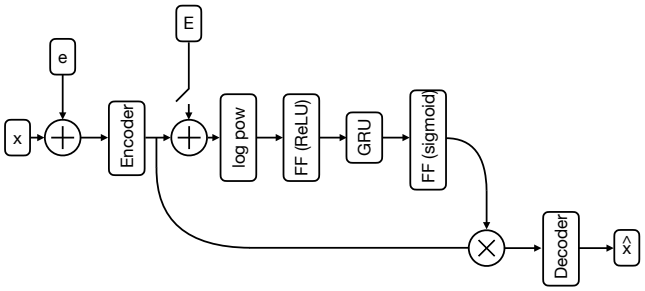

Figure 1: For the implementation of the denoising model we use an encoder–mask–decoder architecture in the low-parameter regime ($\approx 460k$ parameters). Noise e is always present in the input, and in an extended experimental setting, noise E is additionally introduced in the encoder. The encoder is a conv1d layer with 128 filters of length 32 and the decoder its transpose.

3.2. κ-Penalization

To tackle Problem 2) we propose to enforce the trainable filterbank encoder to keep the tight frame property by adding a term to the objective function that penalizes a large condition number. Let $\mathscr{L}(\mathscr{M};x)$ denote any differentiable objective function that is parametrized by the weights of the model $\mathscr{M}$ for an input x, and perform empirical risk minimization with the modified objective

$$\mathscr{L}_\beta(\mathscr{M};x) = \mathscr{L}(\mathscr{M};x) + \beta\cdot\kappa. \tag{6}$$

The hyperparameter β controls the importance of the penalization of large κ to the learning problem. The gradient of κ with respect to the filter entries of the encoder $\phi_j[k]$ is found straightforwardly on the Fourier side. This becomes clear in the next section.

3.3. Fast Computation and the Gradient of κ

The computation of $(\Phi^\top\Phi)^{-\frac{1}{2}}$ and κ can be done efficiently using the FFT paradigm. Denoting by $\widehat{\phi}_j$ the discrete Fourier transforms (DFT) of the filters ϕ_j (which are zero-padded to have the same length as the input), then $\Phi^\top\Phi$ is diagonalized as $\Phi^\top\Phi = U^*\Sigma U$, where $\Sigma = \operatorname{diag}\left(\sum_{k=1}^J |\widehat{\phi}_k|^2\right)$ and U is the unitary DFT matrix. This makes the computation of $(\Phi^\top\Phi)^{-\frac{1}{2}}$, and with that the construction of the nearest Parseval filterbank $\Phi^\sharp$ from Lemma 1 very easy and computationally efficient.

For an efficient computation of κ note that the optimal bounds A, B in Equation 2 are given by the smallest and largest eigenvalue of $\Phi^\top\Phi$, which coincide with those of Σ since U is unitary. Hence [19], they are given by

$$A = \min_{0\le k\le N-1}\sum_{j=1}^J |\widehat{\phi}_j[k]|^2, \qquad B = \max_{0\le k\le N-1}\sum_{j=1}^J |\widehat{\phi}_j[k]|^2. \tag{7}$$

From Equation 7, it is also possible to deduce an explicit expression for the gradient of κ. It is well-defined if the filterbank forms a frame. Using FFT methods we can compute κ and its gradient fast enough to include it in the iterative procedure of gradient descent without significant loss of speed.

4. NUMERICAL EXPERIMENTS ON DENOISING

We apply the proposed stabilization scheme of tightening and κ-penalization in a denoising task using a encoder–mask–decoder architecture. In the following, we describe the experimental setup.

4.1. Dataset

We use the UTD North Texas Vowel Database for the clean speech samples [20]. It consists of recordings from ten male, ten female, and 30 children containing the 12 monophthongal vowels

Model	Objective	SNR [dB]	κ
Tight conv1d	- SNR + 0.5κ	**5.58**	**1**
Conv1d	- SNR	2.61	5.06
Tight conv1d + noise	- SNR + 0.5κ	**0.81**	**1**
Conv1d + noise	- SNR	-0.25	4.98

Table 1: Denoising benchmark on NTVOW with Gaussian noise. For "+ noise", there is Gaussian noise additionally in the encoder. The SNR [dB] is computed on the validation set, and for the condition number of the encoder $\kappa = 1$ is optimal.

present in American English. In total, there are 3190 samples with a length of around one second. As a processing step, we downsample the recordings from the original 48 kHz to 16 kHz. For every speech sample, we add white Gaussian noise with a signal-to-noise ratio (SNR) ranging from -6 to 9 dB in randomly chosen steps of 1 dB. The target signals are the original clean speech signals.

4.2. Encoder/Decoder Design

We initialize a trainable filterbank encoder (conv1d) with 128 filters of length 32 at random and tighten it according to Lemma 1. We set the decoder to be the transposed filterbank of the encoder and do not optimize it during training, i.e., the weights are shared. If the proposed stabilization mechanism yields an approximately tight encoder, the decoder corresponds closely to the dual of the encoder at every step of training.

4.3. Mask Model

Based on the log magnitude responses of the encoder, the model estimates a mask that is applied to the encoder filterbank responses before being decoded. Our model is a simplified version of the architecture proposed in [21]. See Figure 1 for a schematic description. The mask model consists of a feedforward layer (FF) with ReLU activation, a GRU layer, and another feedforward layer with sigmoid activation. It has 460.672 trainable parameters and, hence, plays in the low-parameter regime.

4.4. Training

As a learning objective we use the negative SNR of the denoised signal $\mathscr{M}(x)$, computed by $\mathrm{SNR}(x, x - \mathscr{M}(x)) = \log(\|x\|/\|x - \mathscr{M}(x)\|)$, and modified to penalize a large condition number κ according to Equation 6. This gives the objective

$$\mathscr{L}_\beta(\mathscr{M}; x) = -\mathrm{SNR}(x, x - \mathscr{M}(x)) + \beta \cdot \kappa. \tag{8}$$

We choose $\beta = 0.5$. As an extension, we introduce Gaussian noise to the encoder with zero mean and a variance uniformly chosen between 10^{-3} and 10. This results in an average SNR of the encoder and the noise approximately between -2 and 2 dB. As optimizer, we rely on ADAM with a learning rate of 10^{-5} using 90% of the dataset chosen at random. Validation is done on the residual 10% every 10 epochs. The batch size is set to 16.

5. DISCUSSION AND CONCLUSION

Overall, we can observe from Figure 2 that the proposed stabilization scheme (tight initialization and κ-penalization) for the trainable filterbank encoder works as expected in both settings (with and without noise in the encoder): It keeps the encoder perfectly tight throughout the whole training procedure ($\kappa \approx 1.00026$). If the training is done naively, i.e., without stabilization, the

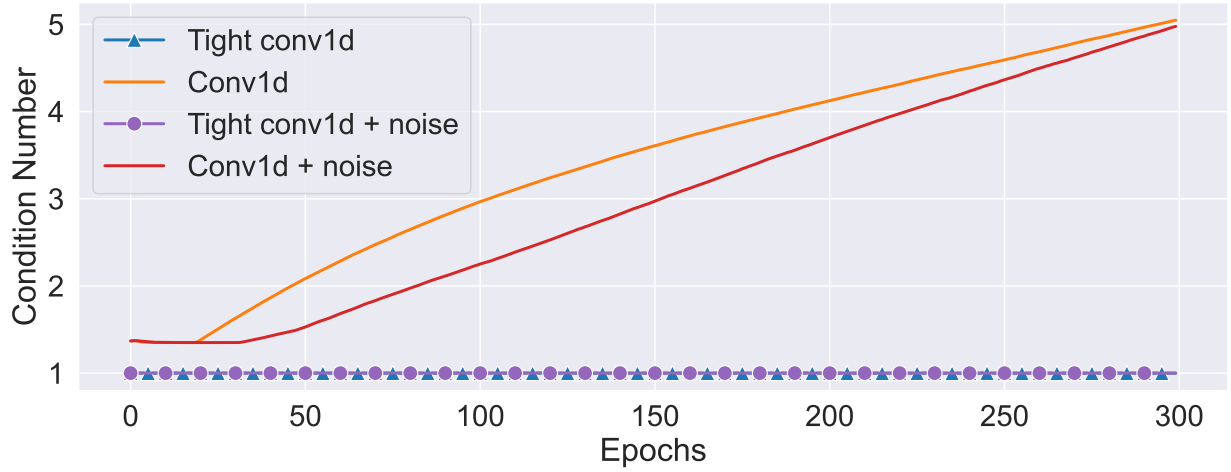

Figure 2: Condition numbers of the encoders for all models are depicted per epoch. Using the proposed stabilization scheme keeps the condition number successfully at one all the time, i.e., the encoder stays tight. The condition number for non-stabilized encoders increases gradually.

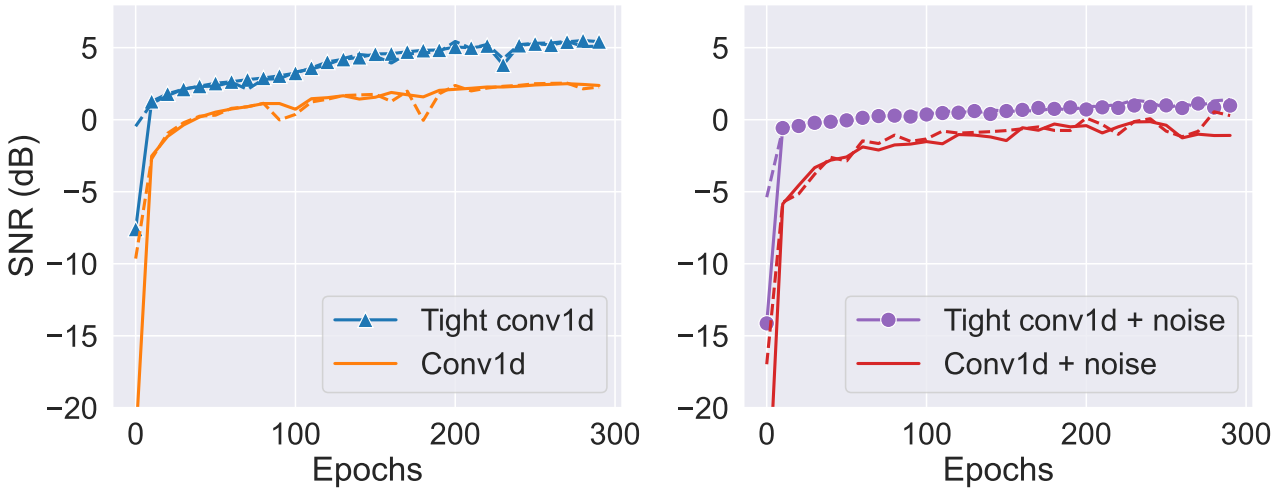

Figure 3: The mean SNR values of the denoised signals on the training (solid) and validation set (dashed) are plotted per epoch. We added markers for the models with tight encoders. Left: No noise in the encoder. Right: Noise in the encoder. In both settings, the models with the stabilized encoders yield higher SNRs of the denoised signals. In the noisy setting (right), the SNR progression of the model with the tight encoder is significantly more stable than the one with the non-stabilized encoder.

condition numbers increase continually (almost linearly) up to $\kappa \approx 5$. We note that a condition number of 5 is in general not considered to be particularly bad, however, it indeed seems to have a large influence on the performance of the denoising model.

From Figure 3 it becomes obvious that the models with the tight encoders reach a significantly higher SNR of the denoised signals than those without stabilization, with a difference of about 3 dB in the last epoch! We can further see that the tight initialization seems to play an essential role in increasing the SNR particularly fast within the first few epochs. Furthermore, also the curves of the SNR progression during training appear to be slightly smoother in the stable setting, which is a sign of more stable training dynamics. This is especially noticeable when noise is in the encoder. In general, the performance in the encoder-noise setting was overall worse that without, but the condition number seems to be affected positively. This needs further investigation.

In summary, we motivated theoretically that tight encoder representations are more robust against noise in the input, and that the reconstruction using the transposed encoder (which equals its pseudo inverse) is more robust against noise in the encoder representation. Based on these results, we made the assumption that tightness should be also beneficial in the deep learning paradigm where noise is present. Indeed, we could show that audio signal denoising significantly benefits from the tightness of the encodings that are used to learn a mask in terms of higher SNRs of the denoised signals. The scheme to implement the stabilization is based on standard results from frame theory and is straightforward and computationally efficient when making use of FFT methods. This positive outcome of this first experiment on tight trainable encoders that are exposed to Gaussian noise triggers a few questions. 1) Does the observed benefit extend to a real-world setting, where background noise is not stationary anymore? 2) With the theory of frame multipliers, can we prove also theoretically that the SNR of the denoised signals is higher if the encoder is tight? 3) How does gradient descent interact with tightness? What is a natural choice for β? Why does the condition number notoriously increase if no stabilization is done? These and further questions will be explored in future work.

ACKNOWLEDGMENT

D. Haider is recipient of a DOC Fellowship of the Austrian Academy of Sciences at the Acoustics Research Institute (A 26355). The work of V. Lostanlen was supported by ANR project MuReNN (ANR-23-CE23-0007-01). The work of P. Balazs was supported by the projects LoFT (P 34624), NoMASP (P 34922), Voice Prints (P 36446) and ChaMp (P 35846) of the Austrian Science Fund (FWF).

REFERENCES

1. P. Mowlaee, J. Kulmer, J. Stahl, and F. Mayer, *Single channel phase-aware signal processing in speech communication: theory and practice*. John Wiley & Sons, 2016.
2. G. S. Moschytz and M. Hofbauer, "Adaptive filter eine einführung in die theorie mit aufgaben und matlab-simulationen auf cd-rom."
3. P. Ochieng, "Deep neural network techniques for monaural speech enhancement and separation: state of the art analysis," *Artif. Intell. Rev.*, vol. 56, no. Suppl 3, p. 3651–3703, oct 2023.
4. H. Dubey, A. Aazami, V. Gopal, B. Naderi, S. Braun, R. Cutler, H. Gamper, M. Golestaneh, and R. Aichner, "Icassp 2023 deep noise suppression challenge," in *ICASSP*, 2023.
5. H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, and T. Sainath, "Deep learning for audio signal processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 206–219, 2019.

6. Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 27, no. 8, p. 1256–1266, 2019.
7. O. Christensen, *An Introduction to Frames and Riesz Bases*, ser. Applied and Numerical Harmonic Analysis. Birkhäuser Boston, 2002.
8. V. K. Goyal, J. Kovačević, and J. A. Kelner, "Quantized frame expansions with erasures," *Applied and Computational Harmonic Analysis*, vol. 10, no. 3, pp. 203–233, 2001.
9. M. Fickus and D. G. Mixon, "Numerically erasure-robust frames," *Linear Algebra and its Applications*, vol. 437, no. 6, pp. 1394–1407, 2012.
10. M. Cisse, P. Bojanowski, E. Grave, Y. Dauphin, and N. Usunier, "Parseval networks: Improving robustness to adversarial examples," *Proc. ICML*, 2017.
11. N. Wiener, *Extrapolation, interpolation, and smoothing of stationary time serie*, J. W. . Sons, Ed., 1949.
12. D. Wang and G. J. Brown, *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Wiley-IEEE Press, 2006.
13. D. S. Williamson and D. Wang, "Time-frequency masking in the complex domain for speech dereverberation and denoising," vol. 25, no. 7, pp. 1492–1501.
14. G. Yu, S. Mallat, and E. Bacry, "Audio denoising by time-frequency block thresholding," *IEEE Trans. on Signal Processing*, vol. 56, no. 5, pp. 1830–1839, May 2008.
15. A. Nicolson and K. K. Paliwal, "Deep learning for minimum mean-square error approaches to speech enhancement," *Speech Communication*, vol. 111, pp. 44–55, 2019.
16. I. J. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
17. G. Matz and F. Hlawatsch, *Linear Time-Frequency Filters: On-line Algorithms and Applications*. eds. A. Papandreou-Suppappola, Boca Raton (FL): CRC Press, 2002, ch. 6 in 'Application in Time-Frequency Signal Processing', pp. 205–271.
18. P. Balazs, "Basic definition and properties of Bessel multipliers," *J. Math. Anal. Appl.*, vol. 325, no. 1, pp. 571–585, January 2007.
19. P. Balazs, N. Holighaus, T. Necciari, and D. Stoeva, *Frame Theory for Signal Processing in Psychoacoustics*. Springer International Publishing, 2017, pp. 225–268.
20. P. F. Assmann and W. F. Katz, "Time-varying spectral change in the vowels of children and adults," *The Journal of the Acoustical Society of America*, vol. 108, no. 4, pp. 1856–1866, 10 2000.
21. S. Braun and I. Tashev, "Data augmentation and loss normalization for deep noise suppression," in *International Conference on Speech and Computer*. Springer, 2020, pp. 79–86.