LEARNING FILTERBANKS FOR END-TO-END ACOUSTIC BEAMFORMING

Samuele Cornell[1]*, Manuel Pariente*[2]*, François Grondin*[3]*, Stefano Squartini*[1]

[1]Università Politecnica delle Marche, Italy
[2]Université de Lorraine, CNRS, Inria, LORIA, France
[3]Université de Sherbrooke, Canada

ABSTRACT

Recent work on monaural source separation has shown that performance can be increased by using fully learned filterbanks with short windows. On the other hand it is widely known that, for conventional beamforming techniques, performance increases with long analysis windows. This applies also to most hybrid neural beamforming methods which rely on a deep neural network (DNN) to estimate the spatial covariance matrices. In this work we try to bridge the gap between these two worlds and explore fully end-to-end hybrid neural beamforming in which, instead of using the Short-Time-Fourier Transform, also the analysis and synthesis filterbanks are learnt jointly with the DNN. In detail, we explore two different types of learned filterbanks: fully learned and analytic. We perform a detailed analysis using the recent Clarity Challenge data and show that by using learnt filterbanks it is possible to surpass oracle-mask based beamforming for short windows.

Index Terms— acoustic beamforming, end-to-end learning, source separation, speech enhancement, multi-channel processing.

1. INTRODUCTION

Most current deep learning based beamforming or *neural beamforming* can be divided into two main categories: *hybrid* [1–11] and *fully neural* [12–16]. Hybrid techniques couple Deep Neural Network (DNN) with established beamforming methods such as Minimum Variance Distortionless Response (MVDR) [17], Multichannel Wiener Filter (MWF) or Generalized Eigenvalue Decomposition (GEV) [18] solutions. Usually they employ a DNN to estimate the spatial covariance matrix (SCM) via a time-frequency mask [1–7] in the magnitude Short-Time Fourier Transform (STFT) domain. Another approach [8] is to use the DNN model to estimate the target and interferer time domain signals and subsequently derive the SCMs. In both cases the DNN is usually a monaural model and the mask is estimated on one microphone channel used as a reference. Additional spatial features are sometimes used to improve the masks estimation [9, 10]. As they rely on SCM estimation to derive the beamforming solution, hybrid neural beamformers performance is greatly affected by the frame size used to estimate the SCMs of the target and interferer/noise signals.

On the other hand, fully neural models employ a DNN to directly estimate the beamforming filters [12, 13] or even the time domain target signal directly [14, 19, 20]. Being fully data-driven these methods are less sensitive to the frame size of the beamforming filters. For example FasNet [12] is able to reach comparable or superior performance with respect to conventional oracle beamformers for low latency applications with remarkably smaller frame size and latency. This is aligned with results in monaural source separation, where fully learned representations have been shown to surpass the STFT in both clean [21] and noisy conditions [22] especially for short windows [23]. However, fully neural models are arguably "less interpretable" and are prone to introducing more nonlinear distortion than conventional beamformers. FasNet [12] is a notable exception as it estimates linear spatial filters for filter-and-sum beamforming, thus enabling to e.g. visualize the beam-patterns. This however is not possible for other methods [14, 19, 20] as the multi-channel processing is done inside the DNN.

In this paper we attempt to bridge the gap between these two paradigms and study conventional beamforming with fully learned filterbanks; inspired by aforementioned promising results achieved in monaural source separation [21–23]. We propose to train a hybrid neural beamformers where the DNN is used to estimate the SCMs via a mask. Unlike previous works [1–11] we learn the analysis and synthesis filterbanks in place of the STFT along with the mask-estimation DNN using time-domain losses. We consider for this study fully unconstrained linear filterbanks as used in [21] and the recently proposed learnable analytic filterbanks [22] which allow for shift invariance, an especially desirable property in this case.

Our findings suggest that, even with a simple hybrid neural beamforming model, performing MVDR and MWF in a learned representation can achieve comparable or superior denoising performance than oracle STFT based beamformers. In detail, regarding MVDR we show that learned filterbanks are consistently able to provide better Scale Invariant Signal-to-Distortion Ratio (SI-SDR) improvement over STFT-based models and, even, oracle Wiener-Like Mask (WLM) [24] for both small and large analysis windows. Notably, the best learned filterbanks model outperform the best oracle WLM configuration by more than 2 dB. Instead, regarding MWF, we show that a significant performance gain over oracle WLM can be achieved for small windows. For longer windows the oracle WLM fares better. In general we found learned filterbanks models to be superior to STFT-based non-oracle ones with the ones based on analytic filterbanks performing the best. We make our code publicly available through the Asteroid Source Separation toolkit [25] [1].

2. ACOUSTIC BEAMFORMING WITH LEARNED FILTERBANKS

Considering an array of M microphones we can denote with $\boldsymbol{y}(t) = [y_1(t), y_2(t), \dots, y_M(t)]^T$ the matrix of the time-domain signals at each microphone, with t being the sample index. We consider here a situation where $\boldsymbol{y}(t)$ is comprised of two terms:

$$\boldsymbol{y}(t) = \boldsymbol{x}(t) + \boldsymbol{\nu}(t), \tag{1}$$

with $\boldsymbol{x}(t)$ the matrix of desired source signals and $\boldsymbol{\nu}(t)$ the matrix of interfering source signals at the microphones. Our goal here

[1]github.com/asteroid-team/asteroid

is recovering the desired signal $x_r(t)$ at an arbitrarily chosen reference microphone $1 \leq r \leq M$ by suppressing the interferer. This implies that, in this study, the target is a reverberated source signal and joint enhancement and dereverberation is left for future work. Accordingly, the target signal at reference microphone r is given by $x_r(t) = \sum_{\tau=1}^{L_h} h_r(\tau) x^a(t-\tau)$, where $x^a(t)$ is the dry desired source signal and $h_r(\tau)$ is the impulse response of length L_h characterizing the acoustic propagation of the desired source signal to the reference microphone at time lag τ. Recovering of $x_r(t)$ can be achieved by conventional spatial filtering techniques if an estimate of the target signal and the interferer SCMs can be produced.

In this paper we follow a simple hybrid neural beamforming framework, illustrated in Figure 1, where such estimates are produced by a monaural mask estimation DNN $\mathcal{F}(\cdot, \boldsymbol{\theta})$ with $\boldsymbol{\theta}$ trainable parameters. An STFT analysis filterbank $\boldsymbol{\phi}_n(t)$ is used to extract the time-frequency representation for every m-th microphone input signal, obtaining a third order tensor:

$$Y_m(n,k) = \sum_{t=1}^{L} y_m(t+kH)\phi_n(t), \;\; n \in [1,\ldots,N], \quad (2)$$

where $\{\phi_n(t)\}_{n=[1,\ldots N]}$ are the N STFT analysis filters each of size $L = N$ and H is the hop-size or stride factor. Consequently n and k denote respectively the frequency bin and frame indexes.

The mask-estimation DNN is given as input features this complex STFT representation (real and imaginary part) at a chosen reference channel r and outputs a mask $m(n,k)$ for the target signal in the magnitude STFT domain (i.e. with shared values between real and imaginary parts):

$$m(n,k) = \sigma(\mathcal{F}(Y_{m=r}(n,k), \boldsymbol{\theta})) \quad (3)$$

where $\sigma(\cdot)$ denotes the sigmoid activation. The interferer signal mask is obtained simply as $1 - m(n,k)$. We found this configuration to work the best in our experiments rather than outputting two distinct masks and/or using a different activation (e.g. softmax). More in detail, this configuration led to more stable training with the dataset used in our experiments, while the use of two distinct masks often led to ill-conditioned SCMs especially when the activation was unbounded (e.g. ReLU).

These masks are then used to compute the frame-wise SCMs of target and interferer respectively:

$$\begin{aligned} \boldsymbol{R}_x(n,k) &= \boldsymbol{Y}(n,k) m(n,k) \boldsymbol{Y}(n,k)^H, \\ \boldsymbol{R}_\nu(n,k) &= \boldsymbol{Y}(n,k)(1-m(n,k))\boldsymbol{Y}(n,k)^H, \end{aligned} \quad (4)$$

where H denotes the Hermitian transpose and both target and interferer SCMs are 4-th order tensors $\in \mathbb{C}^{M\times M\times N\times K}$, where K is the number of frames. In this study, for simplicity, we consider non-causal systems. In this instance, following previous works [1–8], the overall SCM can be computed by simply averaging the frame-wise SCM over the whole input mixture segment: $\boldsymbol{R}_\rho(n) = \frac{1}{K}\sum_k \boldsymbol{R}_\rho(n,k)$ for both target $\rho = x$ and interferer $\rho = n$. In addition to non-causality, this averaging operation requires that the transfer functions of the target source and interferer do not change in the time-frame over which the averaging is performed (e.g. for the target, h_r is assumed stationary).

From such estimated SCMs different beamforming solutions can be computed. In this study we consider MVDR and MWF.

Regarding MVDR, we use the formulation from [26] and estimate the spatial filter as

$$\boldsymbol{w}_m^{MVDR}(n) = \frac{\boldsymbol{R}_\nu^{-1}(n)\boldsymbol{R}_x(n)\boldsymbol{u_m}}{tr\left\{\boldsymbol{R}_\nu^{-1}(n)\boldsymbol{R}_x(n)\right\}}, \quad (5)$$

where $tr\{\cdot\}$ denotes the trace operator and $\boldsymbol{u_m}$ is an one-hot column vector for which the m-th term is 1, and all others are 0. Regarding MWF instead we simply obtain the filter coefficients as:

$$\boldsymbol{w}_m^{MWF}(n) = \left(\frac{1}{\boldsymbol{R}_x(n)+\boldsymbol{R}_\nu(n)}\boldsymbol{R}_x(n)\right)\boldsymbol{u_m}, \quad (6)$$

and the beamformed signal is obtained as

$$\tilde{X}(n,k) = \boldsymbol{w}_m(n)^H \boldsymbol{Y}(n,k), \quad (7)$$

which is finally brought back to time-domain via a *synthesis* inverse-STFT (iSTFT) filterbank $\boldsymbol{\psi}_n(t)$ filterbank with N synthesis filters $\{\boldsymbol{\psi}_n(t)\}_{n=1,\ldots N}$ of, again, length $L = N$ each:

$$\tilde{x}(t) = \sum_{k=1}^{K}\sum_{n=1}^{N} \tilde{X}(n,k)\psi_n(t-kH). \quad (8)$$

2.1. Learnable Analysis and Synthesis Filterbanks

In this work we propose to replace the STFT and iSTFT filterbanks with learnable linear filterbanks and perform spatial filtering in a learned linear basis. These filterbanks are learnt end-to-end jointly along with the mask-estimation DNN $\mathcal{F}(\cdot, \boldsymbol{\theta})$ as the gradient can be back-propagated from a time-domain loss.

We consider here two types of filterbanks: *free* and *analytic* [22] ($\mathcal{A}$) along with the STFT. In free filterbanks both analysis and synthesis parameters are unconstrained as in [21] with N fully learnable real and imaginary parts. For implementation purposes these are treated separately as $2N$ real filters and the whole filterbank is implemented as a 1D convolutional layer.

On the other hand, learnable analytic filterbanks have half of the filters fully learnable. For example, regarding the analysis filterbank $\{\boldsymbol{\phi}_n(t)\}_{n=1,\ldots N}$, for each of the N filters, the imaginary part is obtained from its real counterpart via the Hilbert transform $\mathcal{H}(\cdot)$:

$$\boldsymbol{\phi}_n^{\mathcal{A}}(t) = \boldsymbol{\phi}_n(t) + j\mathcal{H}(\boldsymbol{\phi}_n(t)). \quad (9)$$

The same is true for the synthesis filterbank $\{\boldsymbol{\psi}_n(t)\}_{n=1,\ldots N}$. Because of this coupling, the modulus of a signal convolved with these learnt filters is invariant to small shifts in time domain, a property shared with the STFT. This property is crucial for the estimation of the SCMs in Equation 4, as the target signal mask is estimated on the reference channel and applied across all microphones.

The use of fully learnable filterbanks in place of the STFT poses some problems regarding the derivation of the SCMs. An implicit assumption for Eq. 4 is that the analysis filterbank used is approximately orthogonal. This condition is commonly referred to as the "narrow-band approximation" and can be satisfied by the STFT because of its approximate orthogonality [27].

Without this assumption, the target and interferer SCMs cannot be reduced to $M \times M$ matrices as in Eq.4, as with no orthogonality of the basis, one must take into account also "inter-frequency" terms. This leads to a block matrix SCM for each frame k that can be partioned as an $N \times N$ block matrix (modeling the inter-frequency interactions) with the (i,j)-th block being a $M \times M$ matrix (modeling the inter-microphone interactions). This increases significantly the computational requirements as e.g. inversion of the full SCM leads to a complexity of $\mathcal{O}(N^3M^3)$ versus $\mathcal{O}(NM^3)$ for a diagonal block SCM.

A straightforward, naive, but efficient approach, is to disregard the contribution of the "inter-frequency" interactions in the SCM derivation also for the learned filterbanks. Since the filterbanks are

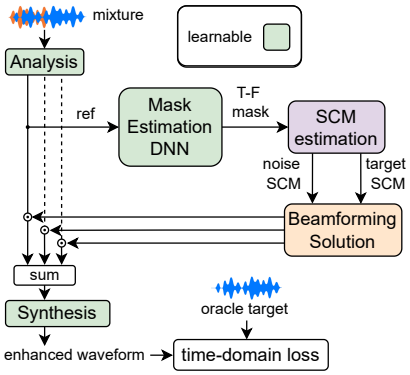

Fig. 1: Framework overview. The gradient is back-propagated from waveform domain. This allows to learn the analysis and synthesis filterbanks along with the mask-estimation DNN.

learnt jointly with the rest of the model by minimizing a particular loss objective (e.g SI-SDR) it can be assumed that the analysis filterbank will learn an approximately orthogonal basis. We adopt in this work this rather strong assumption and provide some empirical evidence for this in Section 4.

3. EXPERIMENTAL SETUP

3.1. Datasets

Crucially, most neural beamforming studies, being targeted mainly towards back-end tasks such as Automatic Speech Recognition (ASR), perform their experiments using 16 kHz signals. Such sampling rate however is sub-optimal for applications aimed towards human listening. For this reason, we use in our experiments the recently available First Clarity Challenge dataset [28] which, being geared towards hearing aid development, is sampled higher at 44.1 kHz.

3.1.1. Clarity Challenge Dataset

We use here the training and development subsets from the Clarity Challenge comprised of, respectively, 6k ($\sim$ 10 h) and 2.5k ($\sim$ 4 h) multi-channel simulated mixtures. Each simulated mixture consists in a target speaker and an interferer signal which can be either another competing speaker or a localized noise source. By dataset construction, each mixture is composed in such a way that the interferer signal always starts 2 seconds before the target signal. To make the task more challenging, in this work we only use 1 second of such "preroll". Spatialization is performed using synthetic Room-Impulse-Responses (RIR) by simulating a randomized room with uniformingly sampled receiver, target and interferer locations, each constrained to be at least 1 m apart from the others. The RIR RT60 has a log-normal distribution with a mean of 0.3 s and a standard deviation of 0.13 s. The Raven toolkit [29] is used to perform such simulation. An array with a behind-the-ear hearing aid topology is employed with 3 microphones per ear. On each ear, microphones are spaced approximately 7.6 mm (front, mid, rear) from one to another. We consider the task of recovering the reverberant target signal at one reference microphone without considering the head related impulse response. In this dataset, the SI-SDR at the array between the target and interferer signals has a -30 to 10 dB range with a skewed gaussian distribution centered around 1 dB. In this work, we report results using the development and use a $90/10$ training set split for the purpose of training and validation respectively.

3.2. Architecture and Training Details

In our experiments we employ ConvTasnet [21] separator as the mask-estimation DNN in Figure 1. We train the whole system comprised of analysis, synthesis mask-estimation DNN and beamforming solution in an end-to-end fashion using negated time-domain SI-SDR [30] as the loss function. Adam [31] is used for optimization along with gradient clipping for gradients exceeding an $\mathcal{L}_2$ norm of 5. We tune learning rate and weight decay for each experiment and train each model for a maximum of 100 epochs with early stopping if no improvement is seen in the last 10 epochs on the validation set. We halve the learning rate if no improvement is seen in the last 5 epochs. During training we randomly choose the reference channel from the 6 available while in testing and validation we always use the first left microphone as the reference.

4. RESULTS

In our experiments we consider, as an upper bound, MVDR and MWF beamformers with oracle WLM in STFT domain. We use as performance metrics SI-SDR improvement (SI-SDRi) and Signal-to-Distortion Ratio [32] improvement (SDRi). The SI-SDR and SDR values for no enhancement are respectively 1.537 dB and 1.144 dB.

In Figure 2a we report the SI-SDRi versus the length of the analysis and synthesis filters (*kernel size*) for different configurations. The number of filters is kept equal to the kernel size, and the stride half of that.

We can see that for both the STFT-based (*STFT*) models and oracle (*oWLM*) masks, performance improves as the kernel size increases. This is expected as a bigger kernel allows for more accurate SCMs estimation. Both *free* and $\mathcal{A}$ learned filterbanks outperform oracle WLM mask for small kernels. Only for MVDR, this is true also for all kernel sizes considered. Interestingly, learned filterbanks seem to have opposing trends regarding MVDR and MWF in function of the kernel size. For MWF performance decreases as the kernel increases.

In Figure 2b we study how SI-SDRi changes by increasing the oversampling factor i.e. decreasing the stride while keeping fixed the kernel size. Here we fix the kernel size and number of filters to 2048 and vary the oversampling factor N/H by 2, from 2 (same as in Figure 2a) to 8.

Regarding MVDR, for both STFT-based systems and *oWLM* performance improves with higher oversampling but at a slower pace compared to what has been observed by increasing kernel size. Regarding MWF, performance decreases slightly for STFT and *oWLM* while is almost constant for the models with learned filterbanks.

In Figure 2c we explore the effect of increasing the number of filters for learned filterbanks with fixed kernel size and stride of respectively 256 and 128 samples. Such strategy is, in fact, one of the key factors which allows current monaural source separation algorithms to achieve such impressive performance [21, 23].

For both beamforming solutions increasing the number of filters and thus forming an over-complete dictionary, improves significantly the performance. By comparing with Figure 2a, we can see that adding filters has a stronger effect with respect to expanding the kernel size. This suggests that beamforming with learned filterbanks may be particularly suited for low-latency applications as the kernel size can be kept low to suit the latency constraints, while the number of filters increased with no penalties in terms of latency.

In Table 1 we compare the best systems from previous experiments (Figure 2) in terms of both SI-SDRi and SDRi. As a term of comparison we also add iFasNet [14], a state-of-the art fully neural beamformer architecture. For this model we use the exact same

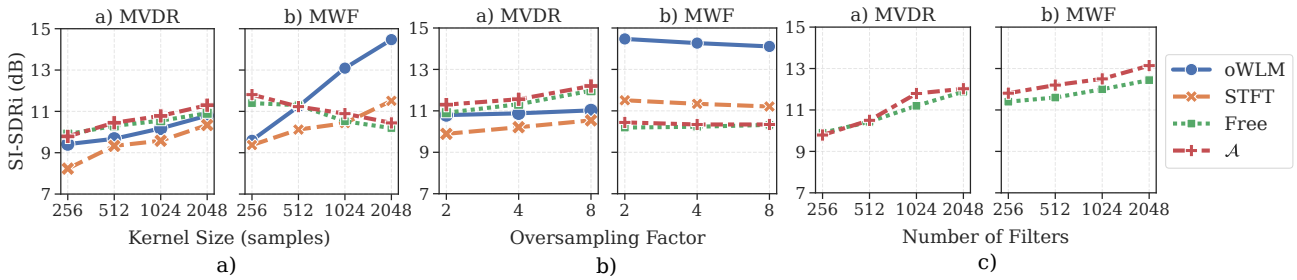

Fig. 2: Performance for different MVDR and MWF configurations: oracle (oWLM) and learned models with different filterbanks (STFT, Free and $\mathcal{A}$). *a)* SI-SDRi versus kernel size. The number of filters is kept equal to kernel size and stride to half. *b)* SI-SDRi versus oversampling factor. The kernel size and number of filters is kept to 2048. *c)* SI-SDRi versus number of filters for learnable filterbanks. The kernel size and stride are kept fixed at 256 and 128 respectively.

Method	N	L	H	SI-SDRi [dB]	SDRi [dB]	Params
oWLM-MVDR	2048	2048	256	11.023	12.410	-
oWLM-MWF	2048	2048	1024	14.733	15.551	-
STFT-MVDR	2048	2048	256	10.321	12.025	5.2M
STFT-MWF	2048	2048	1024	11.556	12.667	5.2M
free-MVDR	2048	256	128	11.882	12.963	6.3M
free-MWF	2048	256	128	12.435	13.632	6.3M
$\mathcal{A}$-MVDR	2048	256	128	12.024	13.372	5.8M
$\mathcal{A}$-MWF	2048	256	128	**13.142**	**14.272**	5.8M
iFasNet [14]	-	-	-	9.896	10.342	4.4M

Table 1: Comparison of best performing models in terms of SI-SDRi and SDRi and number of parameters (*Params.*).

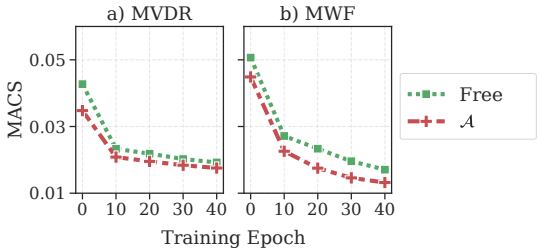

Fig. 3: Mean Absolute Cosine Similarity (MACS) versus training epochs for learned filterbanks (Free and $\mathcal{A}$) MVDR and MWF models. All filterbanks have 1024 filters, 1024 kernel and 512 hop-size.

configuration as in [14]: as the sampling rate here is 44.1 kHz here, iFasNet has more parameters compared to the original one due to increased window length.

The proposed approach is competitive with the current state-of-the-art. Among the non-oracle algorithms, MWF with learned filterbanks obtains the highest figures with the one based on analytic filterbank being the best. This latter consistently surpasses even the best oracle MVDR result.

In Figure 3, we report, at each training epoch, the Mean Absolute Cosine Similarity (MACS) for the analysis filterbanks of MVDR and MWF models with learned filterbanks. In detail, to measure the orthogonality of the learned filterbank, we compute the cosine similarity over each unique pair of filters, take the absolute value and take the average over the total number of unique pairs. We can see that the MACS value decreases during the training, indicating that the analysis filterbank gets more orthogonal as training progresses. This partly confirms the hypothesis made at the end of Section 2.1. On the other hand, the learned filterbanks converge, at best, to a MACS value of 0.013 which is more than one order of magnitude higher than 0.001, obtained for an STFT filterbank with same 1024 kernel size and number of filters. Future work could explore orthogonality constraints and their impact on performance.

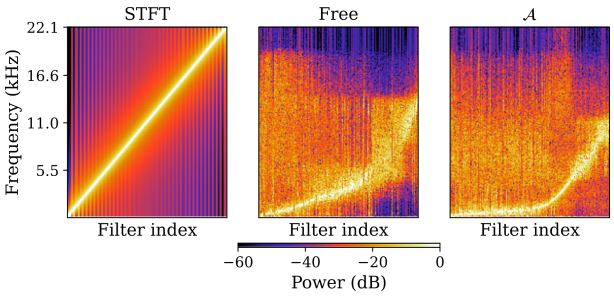

Fig. 4: Frequency response of STFT, free and $\mathcal{A}$ filterbanks. All filterbanks have 2048 filters with 2048 samples kernel size. For visualization purposes, filters in learned filterbanks are sorted according to their center-band frequency.

Finally, in Figure 4 we illustrate the frequency response of STFT and the learned filterbanks under study. Both learned solutions tend to focus more on the lower part of the spectrum where most of speech energy is concentrated. In fact, for free and $\mathcal{A}$, less filters are localized in the higher end of the spectrum, following loosely an exponential trend which is less steep than Mel-scale. This is especially true for $\mathcal{A}$ as most of the filters have a center-band frequency in the sub 2 kHz region leading to an almost piece-wise linear trend. Free filters tend to have an higher frequency spread than analytic ones.

5. CONCLUSIONS

In this work we proposed a fully end-to-end hybrid neural beamforming framework, where a DNN is employed to estimate the SCMs used to derive conventional beamforming solutions such as MVDR and MWF. Differently from previous works, we consider the possibility to learn jointly with the DNN also the analysis and synthesis filterbanks instead of using the STFT and iSTFT. We carried an extensive experimental study comparing learned filterbanks with STFT investigating how performance changes with different kernel sizes, stride factors and number of filters. We found that such proposed strategy of performing spatial filtering in a learned representation is particularly effective for the MVDR beamformer. In fact, in this case we found learned filterbanks to consistently outperform STFT-based ones, even when oracle masks are employed. Regarding MWF, we found out that a gain over oracle masks is possible only for small kernel sizes. This suggests that future work could explore causal, low-latency applications. Among the two learned filterbanks considered, the analytic ones fare the best. This promising result suggests that it may be worth exploring additional inductive biases for learned filterbanks such as orthogonality constraints.

6. REFERENCES

[1] J. Heymann, L. Drude, A. Chinaev, and R. Haeb-Umbach, "Blstm supported gev beamformer front-end for the 3rd chime challenge," in *ASRU*, 2015.

[2] Christoph Boeddeker, Hakan Erdogan, Takuya Yoshioka, and Reinhold Haeb-Umbach, "Exploring practical aspects of neural mask-based beamforming for far-field speech recognition," in *ICASSP*, 2018.

[3] X. Xiao, S. Zhao, D. L. Jones, E. S. Chng, and H. Li, "On time-frequency mask estimation for mvdr beamforming with application in robust speech recognition," in *ICASSP*, 2017.

[4] J. Heymann, L. Drude, C. Boeddeker, P. Hanebrink, and R. Haeb-Umbach, "Beamnet: End-to-end training of a beamformer-supported multi-channel asr system," in *ICASSP*, 2017.

[5] X. Xiao, S. Watanabe, H. Erdogan, L. Lu, J. R. Hershey, M. L. Seltzer, G. Chen, Y. Zhang, M. Mandel, and D. Yu, "Deep beamforming networks for multi-channel speech recognition," in *ICASSP*, 2016.

[6] Hakan Erdogan, John R Hershey, Shinji Watanabe, Michael I Mandel, and Jonathan Le Roux, "Improved mvdr beamforming using single-channel mask prediction networks.," in *INTERSPEECH*, 2016.

[7] T. Ochiai, S. Watanabe, T. Hori, J. R. Hershey, and X. Xiao, "Unified architecture for multichannel end-to-end speech recognition with neural beamforming," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1274–1288, 2017.

[8] T. Ochiai, M. Delcroix, R. Ikeshita, K. Kinoshita, T. Nakatani, and S. Araki, "Beam-tasnet: Time-domain audio separation network meets frequency-domain beamformer," in *ICASSP*, 2020.

[9] Z. Zhang, Y. Xu, M. Yu, S. Zhang, L. Chen, and D. Yu, "Adl-mvdr: All deep learning mvdr beamformer for target speech separation," in *ICASSP*, 2021.

[10] A. Aroudi and S. Braun, "Dbnet: Doa-driven beamforming network for end-to-end farfield sound source separation," *arXiv preprint arXiv:2010.11566*, 2020.

[11] G. Li, S. Liang, S. Nie, W. Liu, Z. Yang, and L. Xiao, "Deep neural network-based generalized sidelobe canceller for robust multi-channel speech recognition.," in *INTERSPEECH*, 2020.

[12] Y. Luo, C. Han, N. Mesgarani, E. Ceolini, and S. Liu, "Fasnet: Low-latency adaptive beamforming for multi-microphone audio processing," in *ASRU*, 2019.

[13] Y. Luo, Z. Chen, N. Mesgarani, and T. Yoshioka, "End-to-end microphone permutation and number invariant multi-channel speech separation," in *ICASSP*, 2020.

[14] Y. Luo and N. Mesgarani, "Implicit filter-and-sum network for multi-channel speech separation," *arXiv preprint arXiv:2011.08401*, 2020.

[15] Y. Xu, Z. Zhang, M. Yu, S. Zhang, and D. Yu, "Generalized spatio-temporal rnn beamformer for target speech separation," *INTERSPEECH*, 2020.

[16] X. Li, Y. Xu, M. Yu, S. Zhang, J. Xu, B. Xu, and D. Yu, "Mimo self-attentive rnn beamformer for multi-speaker speech separation," *INTERSPEECH*, 2021.

[17] J. Capon, "High-resolution frequency-wavenumber spectrum analysis," *Proceedings of the IEEE*, vol. 57, no. 8, pp. 1408–1418, 1969.

[18] E. Warsitz and R. Haeb-Umbach, "Blind acoustic beamforming based on generalized eigenvalue decomposition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1529–1539, 2007.

[19] Jun Qi, Hu Hu, Yannan Wang, Chao-Han Huck Yang, Sabato Marco Siniscalchi, and Chin-Hui Lee, "Exploring deep hybrid tensor-to-vector network architectures for regression based speech enhancement," in *INTERSPEECH*, 2020.

[20] Yihui Fu, Jian Wu, Yanxin Hu, Mengtao Xing, and Lei Xie, "Desnet: A multi-channel network for simultaneous speech dereverberation, enhancement and separation," in *2021 IEEE Spoken Language Technology Workshop (SLT)*, 2021.

[21] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing Ideal Time–Frequency Magnitude Masking for Speech Separation," *IEEE/ACM transactions on audio, speech, and language*, vol. 27, no. 8, pp. 1256–1266, Aug. 2019.

[22] M. Pariente, S. Cornell, A. Deleforge, and E. Vincent, "Filterbank design for end-to-end speech separation," in *ICASSP*, 2020.

[23] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path rnn: efficient long sequence modeling for time-domain single-channel speech separation," in *ICASSP*, 2020.

[24] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *ICASSP*, 2015.

[25] M. Pariente, S. Cornell, J. Cosentino, S. Sivasankaran, E. Tzinis, J. Heitkaemper, M. Olvera, F.-R. Stöter, et al., "Asteroid: the pytorch-based audio source separation toolkit for researchers," *INTERSPEECH*, 2020.

[26] M. Souden, J. Benesty, and S. Affes, "On optimal frequency-domain multichannel linear filtering for noise reduction," *IEEE Transactions on audio, speech, and language processing*, vol. 18, no. 2, pp. 260–276, 2009.

[27] Matthieu Kowalski, Emmanuel Vincent, and Rémi Gribonval, "Beyond the narrowband approximation: Wideband convex methods for under-determined reverberant audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1818–1829, 2010.

[28] M. A Akeroyd, J. P Barker, T. J Cox, J. Culling, S. Graetzer, G. Naylor, E. Porter, and R. Viveros Muñoz, "Launching the first "clarity" machine learning challenge to revolutionise hearing device processing," *The Journal of the Acoustical Society of America*, vol. 148, no. 4, pp. 2711–2711, 2020.

[29] D. Schröder and M. Vorländer, "Raven: A real-time framework for the auralization of interactive virtual environments," in *In Proceedings of Forum Acusticum*, 2011.

[30] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr–half-baked or well done?," in *ICASSP*, 2019.

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[32] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE transactions on audio, speech, and language processing*, vol. 14, no. 4, pp. 1462–1469, 2006.