Bandwidth Extension for Narrowband Speech Signals Based on Beta-Convolutional Non-negative Matrix Factorization

Christian Ryan, Kristian Westerlund and Yu Ge

School of Electrical Engineering and Computer Science
KTH Royal Institute of Technology
cryan@kth.se, kves@kth.se, yuge@kth.se

***Abstract*— In speech coding the main method used to achieve low bitrates is to narrow down the bandwidth of the speech signal. This paper presents two methods to reconstruct the high frequency band of a speech signal. The methods used are convolutional non-negative matrix factorization (CNMF) and non-negative matrix factorization (NMF) and evaluated. Three methods to reconstruct the phase was evaluated with different out comes which is presented in the paper. As a side experiment the Fast Griffin-Lim algorithm was evaluated as well to see how well it reconstructs the phase of a blurred speech signal, though only perceptually.**

I. INTRODUCTION

In speech coding, the main method used to achieve very low bitrates is to narrow down the bandwidth of a speech signal to a low-frequency band, below 4 kHz, which contains the overall information content, like the vocals, consonants, etc. Since the high-frequency band, where many of the fine details of voiced speech lie, is discarded, the speech quality is heavily reduced. One example of this is telephony, or telephone quality speech. This project mainly focuses on recovering the high-frequency band of speech signals using beta-convolutional non-negative matrix factorization (CNMF) to improve audio quality, without sacrificing the data compression benefit of narrowband transmission.

Many reasonable bandwidth extension techniques have been proposed and are still in use today. Traditional solutions rely heavily on the relationship between the observed and missing band of a speech signal to generate a richer reconstruction of the latter. One of the most prominent solutions is called SBR, Spectral Band Replication, described in [1] and [2], which in fact was an integral part behind the industry standard of audio coding, MPEG. The main drawback of this solution is the requirement of a certain amount of side information, such as the speech envelope, harmonics, etc. as described in [2], which diminishes the data compression benefit. However, there exists other methods that do not require any side information. One such technique is the main idea behind this project, which attempts to use a trained codebook or dictionary of typical speech patterns to complete the missing high-frequency band. This method is based on non-negative matrix factorization (NMF), which is the framework this project is based on. Its more powerful extension - convolutional NMF (CNMF) - has also been considered.

In this project, the goal was to extend a narrowband (up to 4kHz) speech signal into a fullband[1] speech signal to improve the quality and mainly focused on two data-driven approaches, NMF and CNMF. Firstly, a codebook was trained for a recorder-specific case by using the NMF algorithm and CNMF algorithm under β-divergence, which is a subclass of Bregman divergence [3]. Then a more general speaker-specific case was studied and lastly a speaker-generic case is discussed in the discussion. Furthermore, three phase reconstruction techniques were implemented: phase lifting, phase mirror and a cross-correlation technique. The Fast Griffin-Lim algorithm (FGLA), a state-of-the-art phase reconstruction technique, was also evaluated using the *ltfat toolbox* [4]. The Fast Griffin-Lim algorithm is an extension of the Griffin-Lim algorithm (GLA) ,which was presented in 1984.

A description of the relevant details regarding the STFT, NMF, CNMF, and β-divergence will be briefly introduced in the Section II. In Section III, the procedure of completing the bandwidth expansion will be presented. The reconstruction results and discussion is presented Section IV and V, respectively.

II. THEORY

A. Short-time Fourier Transform (STFT)

Normally, audio files are analyzed in the frequency domain. Since a sentence is made up of a lot of syllables, it would be beneficial to focus on each syllable of the speech sequence. That means instead of doing a fourier transform on the whole audio file, the short-time Fourier transform (STFT) should be used. The main concept of STFT is to break up whole the data sequence into frames which usually overlap with each other, then do FFT for each frame and the result is added to a matrix, just as in eq. (1).

$$\text{STFT}\{x[n]\} \equiv X(m,w) = \sum_{n=-\infty}^{\infty} x[n]w[n-m]e^{-j\omega n} \quad (1)$$

Where w[n] is the windowing function, in this project a hamming window was used, x[n] is the signal and m denotes the STFT frame/time bin. The STFT is invertible and the original signal can be recovered from the transform by the Inverse STFT (ISTFT) as described in [5].

[1]Original signal was sampled at 16 kHz, i.e. up to 8 kHz.

B. Non-negative Matrix Factorization (NMF)

Non-negative matrix factorization is an algorithm to decompose a matrix V into two matrices W and H as eq.(2). It has been shown that NMF can be used in diverse fields like audio signal processing and computer vision.

$$V \approx W \times H \tag{2}$$

All three matrices do not have any non-negative elements. V, W, H are $M \times N$, $M \times R$, $R \times N$ matrices, respectively. Since each column of W represent a basis vector, W is like a combination of all basis vectors of V and N is determined by the number of basis vectors that V should have, like there are 48 phonemes in English. And the columns of H could be interpreted as the weights which can be used to get the good approximation of V by using the set of the basis vectors W.

C. Convolutional Non-negative Matrix Factorization (CNMF)

If the matrix V is very large, then W and H will naturally too be very large. Then NMF is not suitable anymore, since the far apart elements are generally uncorrelated with another. For example, in a speech segment, the first word seldomly is correlated with the last word in utterance. Therefore a temporal aspect should be taken into account when synthesizing the W and H matrices. Example of this is the convolutional version of NMF, where the bases are not just vectors, but comprise short sequences of vectors, as represented in equation (3).

$$V \approx \sum_{t=0}^{T-1} W_t \times \overset{t\rightarrow}{H} \tag{3}$$

Each W_t is an $M \times R$ non-negative matrix, and V and H are the same as before. The $(t \rightarrow)$ operator is a column shift operator that each column is shifted t places to the right and zeros are filled into the leftmost columns. Conversely, the $(t \leftarrow)$ operator shifts columns off to the left, with zero filling on the right [6].

D. β-Divergence

The factorization usually gets the approximation of V, as shown in equation 2. Therefore, there should be a cost function to evaluate the error between true V and the product of the factorization result $W \times H$. And the cost function is a convex that has the minimum for $V = W \times H$. Then the factorization could be viewed as a minimization problem, as eq.(4).

$$\min_{W,H} C(V \parallel W \times H) = \min_{W,H} D(V \parallel W \times H) \tag{4}$$

subject to W $\geq 0, H \geq 0$, where $D(V \parallel W \times H)$ is the loss function. One of the most popular loss functions is the β-divergence described in [7]. It is a parameterized cost function with a single parameter β, as eq.(5).

$$d_\beta(p \parallel q) = \begin{cases} \dfrac{p^\beta + (\beta-1)q^\beta - \beta p q^{\beta-1}}{\beta(\beta-1)} & \beta \neq 0, 1 \\ p \log \dfrac{p}{q} - p + q & \beta = 1 \\ \dfrac{p}{q} - log\dfrac{p}{q} - 1 & \beta = 0 \end{cases} \tag{5}$$

Then $d_\beta(p \parallel q)$ represents the Itakura-Saito (IS) divergence, the generalized Kullback-Leibler (KL) divergence and the Euclidean distance when $\beta = 0, 1$ and 2 respectively [7]. Then the loss function in eq.(2) under β-divergence could be also defined as eq.(6).

$$D_\beta(V \parallel W \times H) \overset{def}{=} \sum_{m=1}^{M} \sum_{n=1}^{N} d_\beta(v_{mn}, \sum_r m_{mr} h_{rn}) \tag{6}$$

Where $W \times H$ should be replaced with $\sum_{t=0}^{T-1} W_t \times \overset{t\rightarrow}{H}$ in eq.(6), if CNMF is used.

E. Multiplicative Updates Rules

There are many algorithms to do factorization and find good W and H. One of the most prominent solutions is called multiplicative update rule described in [8] which is also the method used in this project.

Firstly the matrices W and H should be initialized as non-negative matrices, and then two matrices are updated by using eq.(7) until they are stable, which is the gradient descent condition for eq.(6). The details of the derivation of eq.(7) can be found in [8].

$$\begin{cases} U = \sum_{t=0}^{T-1} W_t \times \overset{t\rightarrow}{H} \\ w_{t_{k,i}} \leftarrow w_{t_{k,i}} \dfrac{\sum_n v_{k,n} u_{k,n}^{\beta-2} h_{i,n-t}}{\sum_n u_{k,n}^{\beta-1} h_{i,n-t}} \\ h_{i,n} \leftarrow h_{i,n} \dfrac{\sum_{k,t} w_{t_{k,i}} v_{k,n+t} u_{k,n+t}^{\beta-2}}{\sum_{k,t} w_{t_{k,i}} u_{k,n+t}^{\beta-1}} \end{cases} \tag{7}$$

where $w_{t_{k,i}}$ is the k:th row and i:th row component of matrix w_t, similar as matrices H,U,V.

F. Fast Griffin-Lim Algorithm (FGLA)

The GLA projects a signal iteratively onto two different sets in $\mathbb{C}^{\frac{L}{a} \times M}$ denoted by C_1 and C_2.

C_1 is the set of admissible points for eq.(12). It is as well the set of coefficients c that can be reached from $x \in \mathbb{R}^L$ through the frame $\mathbf{G}$ [9]:

$$C_1 = \{c | \exists x \in \mathbb{R}^L \; s.t. \; c = \mathbf{G}x\} \tag{8}$$

The projection can be expressed as:

$$P_{C_1}(c) = \mathbf{G}\mathbf{G}^\dagger c \tag{9}$$

C_2 is the set of coefficients that minimize eq.(12). It is given by

$$C_2 = \left\{ c \in \mathbb{C}^{MN} \;\middle|\; |c| = s \right\} \tag{10}$$

The projection onto C_2 is equivalent to force the magnitude of s to be c element wise [9]:

$$P_{C_2}(c) = s \cdot e^{i \angle c} \tag{11}$$

The problem can be expressed as finding a signal $x^* \in \mathbb{R}^L$ from a given set of non-negative coefficients s, such that the magnitude of the STFT of $x^* : |\mathbf{G}x|$ is as close as possible. The l_2-norm is used to measure the closeness [9]. Classic optimization algorithms depends on the problem being convex, due to the computational cost of the methods makes it unstuitable for longer signals. the idea behind the FGLA is to find the solution of the non convex problem

$$\text{minimize}_{c\in\mathbb{C}^{MN}} \parallel |c| - s \parallel_2 \ \ s.t. \exists x \ \in \mathbb{R}^L \mid c = \mathbf{G}x \quad (12)$$

The FGLA finds the intersection between the two sets C_1 and C_2 so iterative projections would converge to an optimal solution [9].

Fast Griffin-Lim algorithm (FGLA)

Fix the initial phase $\angle c_0$
Initialize $c_0 = s \cdot e^{i\cdot\angle c_0}$, $t_0 = Pc_2(Pc_1(c_0))$
Iterate for $n = 1, 2, ...$
$t_n = Pc_1(Pc_2(c_{n-1}))$
$c_n = t_n + \alpha_n(t_n - t_{n-1})$
Update α_n
Until convergence
$x^* = G^\dagger c_n$

III. PROCEDURE

A. Magnitude reconstruction

To have appropriate data to manipulate, the open-source LibriSpeech ASR corpus was used, available through [10]. These audio files are of high SNR, with a diverse set of different speakers of both genders, however, only in English. Pan-language dictionary learning was beyond the scope of this project. These files were sampled at 16 kHz, rendering a frequency spectrum from 0 to 8 kHz. For the recorder-specific case, merely one audio file was picked at random from the LibriSpeech corpus; meanwhile for the speaker-specific, approximately 15 audio files were concatenated into one, which was used training and lastly one unseen file of the same speaker was kept separate as a test set. With this audio file, either single or concatenated, a spectrogram could be obtained by using the STFT technique explained in II-A.

For the overlap and add STFT analysis, a hamming window with 75 percent overlap was used, as this would yield perfect reconstruction[2] according to [11]. As audio with sampling rate of 16 kHz was used, an appropriate window length of $M = 1024$ was necessary, as this rendered a frequency resolution of approximately 16 Hz per frequency bin as well as a temporal resolution of approximately 16 ms per time bin, which matches the limit to perceptual distinguishability of human hearing according to [12].

As per eq.(5), there are a lot of different loss functions that could be appropriate for this project's use case, such as the mean square error ($\beta = 2$) or the KL divergence ($\beta = 1$). However, studies in [13] and [14] have shown that $\beta \approx 0.5$ is suitable for similar use cases. This β-value was evaluated in section IV.

With the above framework in mind, the spectral magnitude bases matrix, W, can be trained using the NMF or consequently CNMF training algorithms. In either case, the magnitude spectrogram, V in II-B, can be extracted from a dataset[3], by taking the absolute value of its STFT. When the NMF/CNMF procedure converges under beta-divergence, then the training of the fullband W, W_{FB}, is complete. Thereafter, W_{FB} needs to be cut appropriately to match the STFT dimensions of the narrowband of the test signal, which in this project used a cutoff frequency of 4 kHz, i.e. half of the fullband, W_{HB}. With W_{HB}, H can be extracted using the same updates and divergence as for W_{FB}, i.e. eq.(7), under the strict condition that W does not update concurrently, as W_{FB} should not be change after training. Afterwards, the update H should be multiplied with W_{FB}, or convolutionally summed for CNMF, as per section II-D to obtain the estimated spectrogram $\hat{V}$. However, as the narrowband is already transmitted, this should replace the narrowband of the estimate $\hat{V}$ to improve the reconstruction.

B. Phase reconstruction

For the phase reconstruction, three different techniques were implemented, two somewhat straightforward and one more nuanced. The initial technique, appropriately named phase lifting, involves merely copying the phase information of the narrowband into the high-frequency band, i.e. lifting the phase of the narrowband up, such that a fullband spectrum is obtained and the ISTFT can be performed. This technique is crude, but valuable as a baseline.

For a smoother phase transition around the cutoff frequency, the phase information of the narrowband can instead be mirrored around the cutoff frequency into the high-frequency band. This will yield a phase reconstruction that is smoother around the cutoff frequency compared to abruptly lifting the phase from the low frequencies of the narrowband. This method is denoted as phase mirroring.

Lastly, a method using cross-correlation was implemented. The basic idea is to find the lag that corresponds to the highest correlation between the narrowband and the high-frequency band for each time bin (or SFFT frame) before transmission, and as such an appropriate lifting can be done from the narrowband to the high-frequency band for each individual time bin. Using a similar analysis as for the phase mirroring technique, the phase information can also be mirrored around the point where the lag shift ends, presuming the lifting will deviate from the true fullband size, ensuring a smoother phase reconstruction.

IV. RESULTS

Using a cutoff frequency of $f_c = 4$kHz, 1000 iterations[4] when training W and a convolutional length of $T = 100$ for the CNMF. Only 500 iterations were necessary when

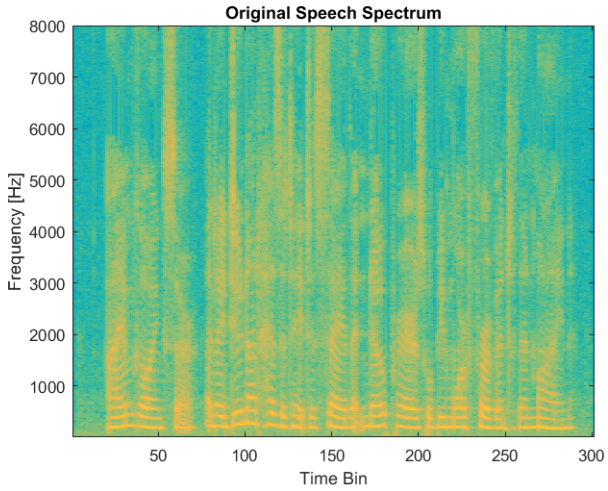
[2]Strictly looking at analysis and synthesis.

[3]Either single audio file or concatenated, depending on recorder- or speaker-specific use case.

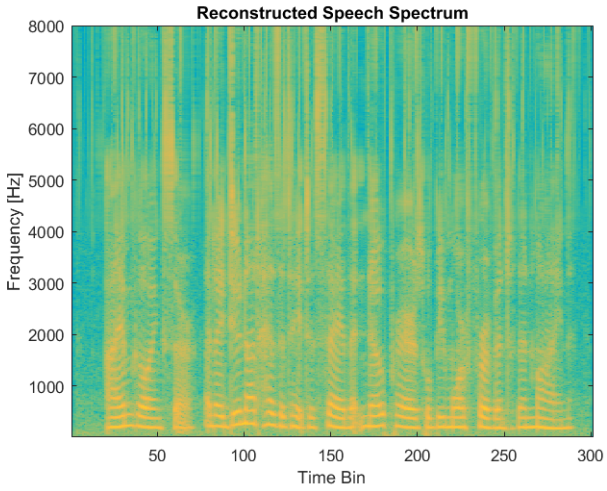
[4]Limited to this value due to computational complexity.

using NMF as it had already converged at that point. For all spectrograms and magnitude related plots, perfect phase reconstruction was assumed, i.e. the original phase information was transmitted and vice versa regarding phase related plots. This was done to accurately be able to compare reconstruction techniques. Analyzing the mean square error (MSE) of the phase reconstruction with the original phase information in the frequency domain would potentially yield misleading results, due to the periodicity of phase, i.e. an RMSE of π might not seem much when regarding a huge matrix but implies the opposite sign in the time domain and equally confusing is an RMSE of 2π. Therefore, to accurately be able to compare the phase reconstruction techniques, the MSE of the time domain signal of the original and reconstruction was regarded instead.

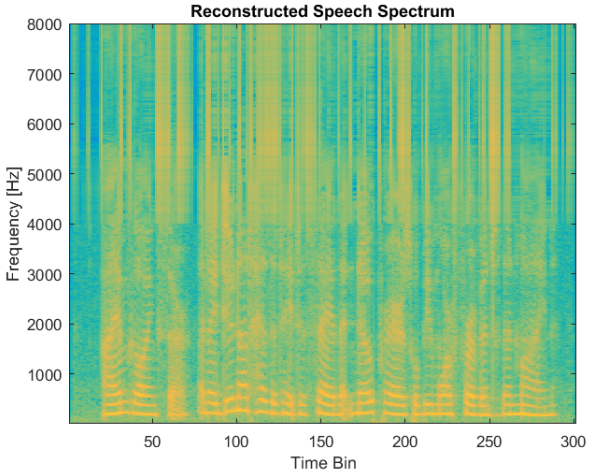
Note that if the phase of the reconstruction does not entirely match the original signal, then complex-values will be obtained when performing the ISTFT. For this reason, the authors of this paper have decided to merely focus on the real part of the ISTFT of the reconstruction, as the imaginary part will naturally decay as the phase reconstruction becomes more accurate.



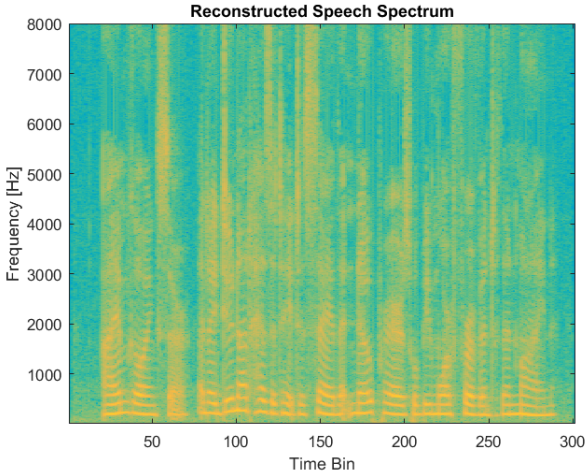
(a) Spectrogram of original speech signal.



(b) Spectrogram of reconstructed speech signal using NMF with $\beta = 0.5$.

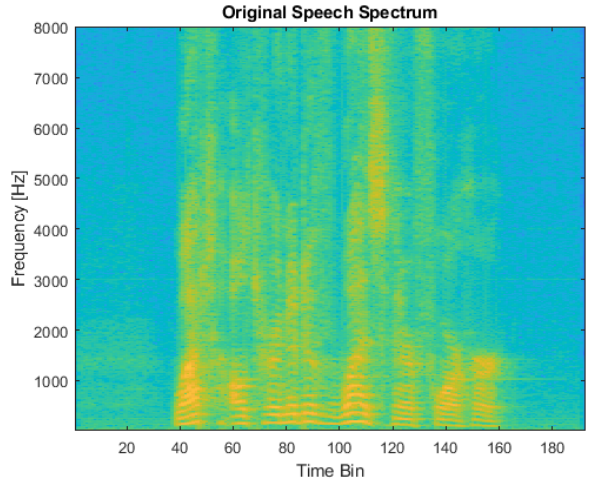


(c) Spectrogram of reconstructed speech signal using NMF with $\beta = 2$.

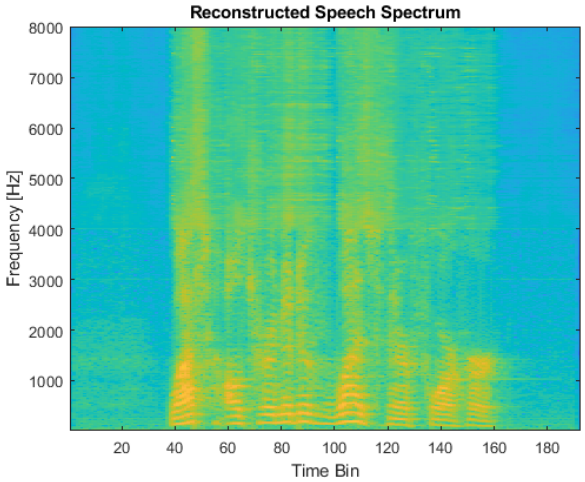


(d) Spectrogram of reconstructed speech signal using CNMF with $\beta = 0.5$.

Fig. 1: Spectrograms for the recorder-specific case with cutoff frequency $f_c = 4$kHz.

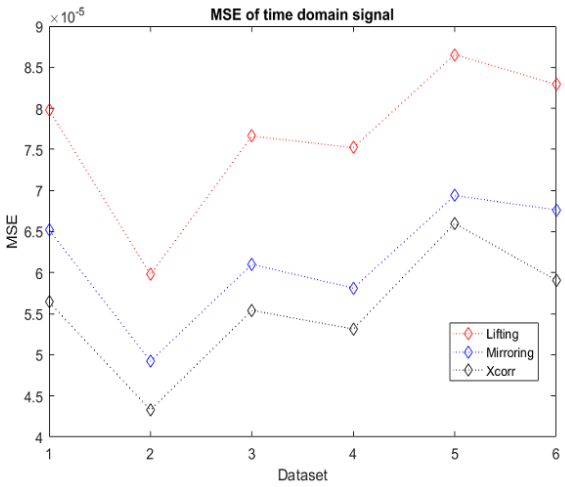


(a) Spectrogram of original unseen speech signal.

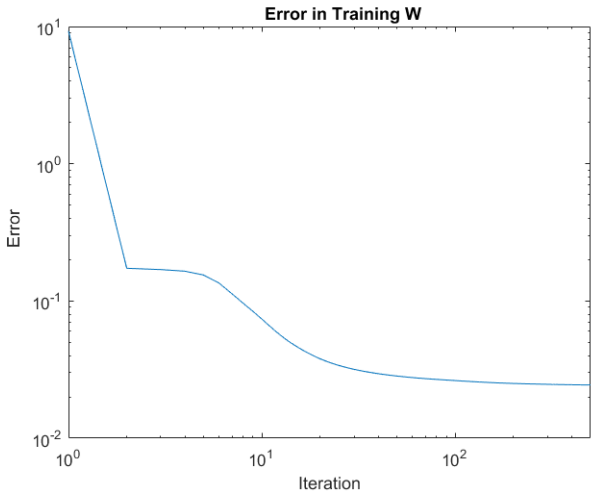


(b) Spectrogram of reconstructed speech signal using CNMF with $\beta = 0.5$, trained under 15 audio files.

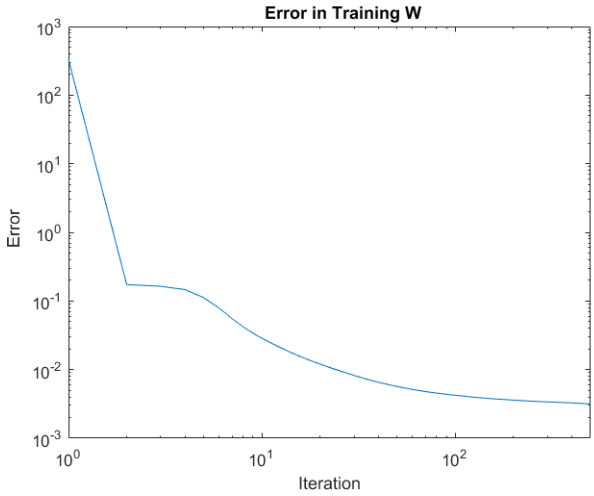
Fig. 2: Spectrograms for the speaker-specific case with cutoff frequency $f_c = 4$kHz.



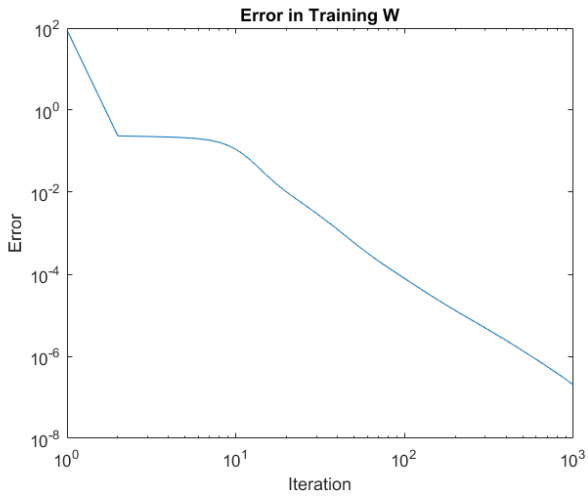
(a) MSE between original signal and reconstructed signal in time domain using different phase reconstruction techniques. X-axis denotes different audio data files picked at random from LibriSpeech.



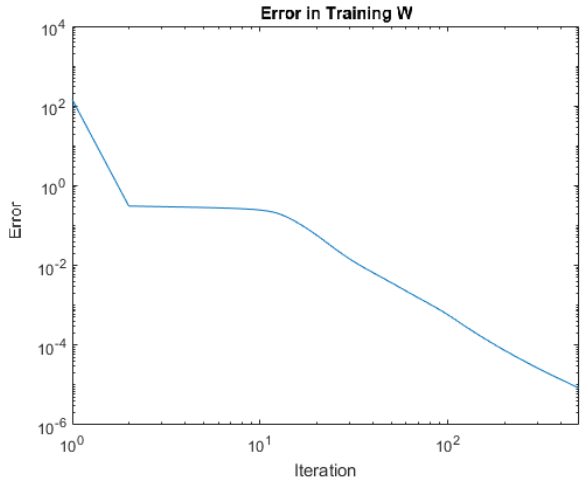
(b) Error when training W for case fig. 1b.



(c) Error when training W for case fig. 1c.



(d) Error when training W for case fig. 1d.



(e) Error when training W for generic case, fig. 2b.

Fig. 3: Errors obtained while training W as well as a comparison of the phase reconstruction techniques.

V. DISCUSSION

A. Magnitude reconstruction

Analyzing fig. 1b and fig. 1c one can immediately note that a $\beta = 0.5$ generates a substantially better reconstruction compared to a Euclidean distance loss function, even though the error when training W is in fact smaller for the latter compared to the former. This illustrates the duality and difficulty of utilizing an appropriate loss function and confirms the findings found in [13] and [14].

Upon inspecting fig. 1d, a near perfect reconstruction is obtained when using CNMF compared to the NMF results in fig. 1b. This is also substantiated by the results obtained in fig. 3d and fig. 3b, yielding a much lower error when training W. This illustrates the power but also the main drawback of the CNMF, that the accuracy is heavily improved but at the cost of computational complexity. In fig. 3d one can note that the error has not had the chance to converge, but was rather terminated due to its long run time. In this recorder-specific case, that is not a problem as the error had already reached a value of 10^{-6}. However, when trying a more general approach with a speaker-specific training set[5] as in fig. 2, the results were far more lackluster, even though the training of the generic case has a small learning error as in fig. 3e. The reconstruction is generic and noisy, exemplified by the smeared out spectrogram from 4 kHz to 8 kHz in fig. 2b and lacks the fine structure of the recorder-specific CNMF reconstruction. The reason for this is the lackluster size of the training dataset, i.e. the codebook was trained on too few speech segments to accurately estimate the unseen sequence. A set of only 15 audio files, on average 8 seconds long and sampled at 16 kHz, yields approximately 2 million data points that need to be analyzed and dissected by the training algorithm. Add the convolutional aspect of $T = 100$ for the CNMF, meaning 100 codebooks of 2 million data points each need to be considered (see eq. (3)), and the computational complexity illustrates itself. However, this computational cost is in matter of fact merely a one-time expense, as when the codebook is trained, it shall remain unchanged. It is for this reason we, the authors, firmly believe in the viability of this technique under the condition that it is trained on far more powerful resources than what we had at our disposal for this project, an Intel-Core i5-4460 CPU @ 3.2 GHz, 4 cores, 8 Gb RAM running windows 10. Use cases like speaker-generic codebook training should in theory merely be an extension of the speaker-specific use case, where even more training data needs to be supplied to cover all the nuances of speech in the given language, furthering the usability of non-negative matrix factorization as tool for bandwidth extension of narrowband speech signals.

B. Phase reconstruction

In fig. 3a the MSE of the time domain signal for each phase reconstruction technique for six different datasets can be seen. These datasets are in fact different audio recordings of the same speaker. It can be noted that the cross-correlation technique explained in section III in fact outperformed the other phase reconstruction techniques. Furthermore, fig. 3a seems to suggest that the cross-correlation technique improves consistently improves the reconstruction by 25% independent of the dataset, although this is a small sample size. The results confirm the nuances taken into account by appropriately shifting the narrowband for each time bin, as explained above. This technique does have a main drawback, however, that calculating the appropriate lag for each time bin means that this information must also be transmitted with the narrowband and thereby side-information is introduced. The amount of side-information also grows with the length of audio sequence, diminishing the data compression benefit. For this reason, the phase mirroring technique seems most viable, as the MSE results do not differ too much from the cross-correlation method and also retains the benefit of no side-information. A possible extension of this project would therefore be to build upon this crude phase mirroring technique for an even better reconstruction, however, at the same time maintaining the no side-information advantage.

C. Fast Griffin-Lim algorithm (FGLA)

The FGLA was another phase reconstruction method evaluated during this project, it is incorporated in the *ltfat toolbox* [4]. Due to the algorithm being incorporated in a Matlab toolbox the methods to reconstruct the amplitude used in this project could not be used. So other means to simulate a signal that has been reconstructed with the help of a codebook needed to be used. A reverb was used to accomplish this, to smear out the amplitude of the signal, i.e. mimicking an imperfect reconstruction. The FGLA was able to reconstruct the phase with perceptually good results, although this was merely tested on one speech sample. This reconstruction was although completed with a completely arbitrary initialization, meaning the knowledge of the narrowband was not even used. This exemplifies the sheer potential of the FGLA algortihm, as it will merely improve with having more information at its disposal. The algorithm might be cumbersome to use in online applications, however, due to the fact that 100 iterations, which was necessary for a perceptually decent reconstruction[6], takes about 10 seconds to complete[7] and the reconstruction sounds decent only after 10 iterations. It is therefore possible to reconstruct the phase only from the amplitude of the signal, thus combining the FGLA with the CNMF training algorithm would be a new method of reconstructing signals with no side-information at all.

D. Optimization possibilities

As noted above, the computational complexity is a large drawback of the CNMF. However, the accuracy in reconstruction far outperforms the NMF - rendering this technique to be the better of the two. One possibility to optimize the learning of the CNMF codebook is to remove any silent

[5]Even though this is far from speaker-generic, it illustrates the same procedure.

[6]According to our subjective hearing.

[7]Using the same equipment described in section V-A.

pauses or unintelligible sound from the training set. By doing so one maximizes the learning capability of each STFT frame, as each time bin contains rich and nuanced speech. Furthermore, there is no point in arduously learning spectral bases for pauses and/or exasperations as the reconstruction is meant for speech not silence.

E. State-of-the-art comparison and possible extensions

The results of this project are in no way in competition with the state-of-the-art techniques currently used today, such as MPEG. However, the findings of this project do suggest the potential viability for a new powerful tool in bandwidth extension field, that being combining the power of CNMF with say the state-of-the-art phase reconstruction technique, the FGLA algorithm, or an improvement of the crude phase mirroring technique presented in this paper. All of this are potential extensions of this project, and something we, the authors, encourage you, the reader, to delve into.

VI. ACKNOWLEDGEMENT

The authors of this paper would like to acknowledge the supervision and support of Dr. Stanislaw Gorlow, without whom this paper would not have been made possible. Furthermore, an acknowledgement of the support, facilities and goodwill of the School of Electrical Engineering and Computer Science is in order.

REFERENCES

[1] V. Britanak, "Spectral band replication (sbr) compression technology: Survey of the unified efficient low-cost implementations of complex exponential- and cosine-modulated qmf banks," *Signal Processing*, vol. 115, pp. 49–65, 2015, ISSN: 0165-1684.

[2] P. Ekstrand, "Bandwidth extension of audio signals by spectral band replication," in *in Proceedings of the 1st IEEE Benelux Workshop on Model Based Processing and Coding of Audio (MPCA)*, Citeseer, 2002.

[3] L. M. Bregman, "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming," *USSR computational mathematics and mathematical physics*, vol. 7, no. 3, pp. 200–217, 1967.

[4] P. L. Søndergaard, B. Torrésani, and P. Balazs, "The Linear Time Frequency Analysis Toolbox," *International Journal of Wavelets, Multiresolution Analysis and Information Processing*, vol. 10, no. 4, 2012. DOI: 10.1142/S0219691312500324.

[5] J. Allen, "Short term spectral analysis, synthesis, and modification by discrete fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 25, no. 3, pp. 235–238, 1977.

[6] P. D. O'grady and B. A. Pearlmutter, "Convolutive non-negative matrix factorisation with a sparseness constraint," in *Machine Learning for Signal Processing, 2006. Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on*, IEEE, 2006, pp. 427–432.

[7] C. Févotte and J. Idier, "Algorithms for nonnegative matrix factorization with the β-divergence," *Neural computation*, vol. 23, no. 9, pp. 2421–2456, 2011.

[8] S. Gorlow *et al.*, "Exact multiplicative updates for convolutional β-nmf in 2d," *arXiv preprint arXiv:1811.01661*, 2018.

[9] N. Perraudin, P. Balazs, and P. L. Søndergaard, "A fast griffin-lim algorithm," in *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 2013, pp. 1–4. DOI: 10.1109/WASPAA.2013.6701851.

[10] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, IEEE, 2015, pp. 5206–5210.

[11] J. O. Smith, *Spectral Audio Signal Processing*. http://-ccrma.stanford.edu/~jos/sasp/, accessed ¡date¿, online book, 2011 edition.

[12] J. N. Holmes, *Speech synthesis and recognition*, eng, 2. ed.. New York ; London: Taylor & Francis, 2001, ISBN: 0-7484-0857-6pbk.

[13] D. FitzGerald, M. Cranitch, and E. Coyle, "On the use of the beta divergence for musical source separation," 2009.

[14] E. Vincent, N. Bertin, and R. Badeau, "Adaptive harmonic spectral decomposition for multiple pitch estimation,"