INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Weighted Von Mises Distribution-based Loss Function for Real-time STFT Phase Reconstruction Using DNN

Nguyen Binh Thien[1], *Yukoh Wakabayashi*[2], *Geng Yuting*[1], *Kenta Iwai*[1], *Takanobu Nishiura*[1]

[1]Ritsumeikan University, Shiga, Japan
[2]Toyohashi University of Technology, Aichi, Japan

gr0398xe@ed.ritsumei.ac.jp, wakayuko@cs.tut.ac.jp, geng@fc.ritsumei.ac.jp,
iwai18sp@fc.ritsumei.ac.jp, nishiura@is.ritsumei.ac.jp

Abstract

This paper presents improvements to real-time phase reconstruction using deep neural networks (DNNs). The advantage of DNN-based approaches in phase reconstruction is that they can leverage prior knowledge from data and are adaptable to real-time applications by using causal models. However, conventional DNN-based methods do not consider the varying properties of the phase at different time–frequency bins. Our paper proposes loss functions for phase reconstruction that incorporate frequency-specific and amplitude weights to distinguish the importance of phase elements based on their properties. We also use an extension of the group delay to improve the phase connections along the frequency. To improve the generalization, we augment the data by randomly shifting the signals in the time domain for each epoch during training. Experimental results show the superior performance of the proposed methods compared to conventional DNN-based and non-DNN real-time phase reconstruction methods.

Index Terms: Deep neural network, phase reconstruction, instantaneous frequency, group delay, von Mises distribution

1. Introduction

Short-time Fourier transform (STFT) phase reconstruction has recently been receiving increased attention [1–18]. In contrast to conventional STFT-based applications, which only process the amplitude, the phase reconstruction can help produce higher quality time-domain signals in various fields, including speech enhancement [8–11] and source separation [3, 4, 12]. However, reconstructing the phase is challenging due to the wrapping issue and phase sensitivity to waveform shift. In addition, when only the amplitude is available, the phase reconstruction is affected by the sign indetermination problem. For example, the STFTs of two opposite signals, $x(n)$ and $-x(n)$, have the same amplitude but different phase spectrograms. Various phase reconstruction methods have been proposed, including those based on STFT consistency [13, 14], models [8, 9], optimization [15, 16], phase gradient heap integration (PGHI) [17], and deep neural networks (DNNs) [1–7, 18].

Many phase reconstruction algorithms require iteration or future frame information to estimate the current-frame phase, which may only be feasible offline. For real-time settings, some modifications have been made. [19] proposed a real-time version of the Griffin–Lim algorithm, called the real-time spectrogram inversion algorithm (RTISI), which iteratively reconstructs the signal frame-by-frame with an effective initialization scheme. In a non-iterative manner, the single-pass spectrogram inversion (SPSI) [20] utilizes a phase-locking technique related to a phase vocoder. [21] proposed a real-time adaptation of the PGHI (i.e., RTPGHI) with one or even zero look-ahead frames. Although these methods have achieved promising results, they are still suboptimal due to some approximations used, e.g., the harmonic model assumption in the SPSI and the phase derivative approximation in the RTPGHI. Among phase reconstruction approaches, DNN-based methods have significant potential for real-time applications, as they can be easily adapted by using a causal model. Additionally, DNNs have a robust modeling capability to learn the underlying structure of the target signals.

However, most of the conventional DNN-based phase reconstruction methods do not consider the distinct properties of the phase at different time–frequency (TF) bins. In the inverse STFT (ISTFT), the amplitude acts as a scaling factor for the TF bins, and the phase determines their relative position, thus ensuring their proper combination. If the amplitude is low, regardless of the values of the phase, the contribution of the TF bin to the reconstructed signal will be small. Training a model with the phase at these low-amplitude bins may not bring much benefit and might even restrict the model from learning useful information in the high-amplitude bins. Another property of the phase is that, unlike the amplitude, its values depend linearly on the frequency. At high frequencies, the phase changes quickly along the time, leading to instability. These unstable phase elements usually yield high errors, which may impede the model from fitting the more stable phase elements at low frequencies.

Taking into account the varying properties of the phase, the aim of this paper is to improve DNN-based methods for real-time phase reconstruction from the amplitude, including proposing new loss functions and data augmentation methods. Starting with the *von Mises* distribution-based loss functions as in [1] and [2], we impose weights on the phase loss with respect to frequency to control the effect of unstable phase elements at high frequencies. We also leverage amplitude weights to separate the importance of the phase at different TF bins. In addition, we extend a phase feature, group delay (GD), and include it in the loss function to improve the connection of phase elements along the frequency. The proposed loss functions are utilized to train a causal DNN architecture for real-time applications. To improve the generalization of the models, we augment the training data by randomly shifting the signals in the time domain before calculating the STFT for each training epoch.

2. Conventional loss functions for DNN-based phase reconstruction

In this section, we define the notation and review two baseline loss functions for DNN-based phase reconstruction. Let $|X_{k,\ell}|$ and $\Phi_{k,\ell}$ represent the STFT amplitude and phase of a discrete-time signal, where $k \in \{0, \dots, K-1\}$ and $\ell \in \{0, \dots, L-1\}$ are the frequency bin index and time frame index, respectively. Their matrix notations are represented by

10.21437/Interspeech.2023-580

bold letters, i.e., $|\boldsymbol{X}|$ and $\boldsymbol{\Phi}$. The STFT is calculated with the window length of M samples, the window shift of R samples, and the number of DFT points of N.

2.1. *Von Mises* distribution-based loss function [1]

To handle the periodicity of the phase, [1] modeled the phase using the *von Mises* distribution, which is a circular distribution. Its probability density function is defined as

$$f(\Phi_{k,\ell}|\mu,\kappa) = \frac{e^{\kappa\cos(\Phi_{k,\ell}-\mu)}}{2\pi I_0(\kappa)}, \tag{1}$$

where μ, κ, and $I_0(\cdot)$ represent a measure of location, a measure of concentration, and the modified Bessel function of order 0, respectively. The negative log-likelihood of (1) is defined as

$$-\log f(\Phi_{k,\ell}|\mu,\kappa) = -\kappa\cos(\Phi_{k,\ell}-\mu) + \text{const.}, \tag{2}$$

where const. is a constant with respect to $\Phi_{k,\ell}$. With the assumption of a constant κ, the phase loss function is defined as

$$\mathcal{L}_{\text{p}}(\boldsymbol{\Phi},\hat{\boldsymbol{\Phi}}) = -\sum_{k,\ell}\mathcal{C}^{\text{p}}_{k,\ell} \triangleq -\sum_{k,\ell}\cos(\Phi_{k,\ell}-\hat{\Phi}_{k,\ell}). \tag{3}$$

To improve the performance, [1] utilized a phase feature, i.e., the GD [22]. The GD is defined as the negative frequency derivative of the phase and is calculated as

$$U_{k,\ell} = \mathcal{P}(\Phi_{k,\ell} - \Phi_{k+1,\ell}), \tag{4}$$

where $\mathcal{P}(\cdot)$ is a function that wraps a value into the principal range of $(-\pi,\pi]$. By modeling the GD with the *von Mises* distribution, [1] introduced a multitask-learning loss function for phase reconstruction, which can be expressed as

$$\mathcal{L}_{\text{pgd}}(\boldsymbol{\Phi},\hat{\boldsymbol{\Phi}}) = -\sum_{k,\ell}\left(\lambda_{\text{p}}\mathcal{C}^{\text{p}}_{k,\ell} + \lambda_{\text{gd}}\mathcal{C}^{\text{gd}}_{k,\ell}\right), \tag{5}$$

where

$$\mathcal{C}^{\text{gd}}_{k,\ell} = \cos(U_{k,\ell} - \hat{U}_{k,\ell}), \tag{6}$$

and λ_{p} and λ_{gd} are the weights for the loss components.

2.2. *Von Mises* mixture model-based loss function [2]

To mitigate the sign indetermination problem, the idea in [2] is to reconstruct the phase of either $x(n)$ or $-x(n)$. The phases of $x(n)$ and $-x(n)$ have a difference of π and can be modeled by a *von Mises* mixture model with two mixture components, as

$$\mathcal{F}(\Phi_{k,\ell}|\mu,\kappa) = \frac{1}{2}f(\Phi_{k,\ell}|\mu,\kappa) + \frac{1}{2}f(\Phi_{k,\ell}|\mu+\pi,\kappa). \tag{7}$$

Assuming $\kappa = 1$, the *von Mises* mixture model-based loss function is defined using maximum likelihood, as

$$\begin{aligned}\mathcal{L}_{\text{vmm}}(\boldsymbol{\Phi},\hat{\boldsymbol{\Phi}}) &= -\sum_{k,\ell}\mathcal{C}^{\text{vmm}}_{k,\ell} \\ &\triangleq -\sum_{k,\ell}\log\left(e^{\cos(\Phi_{k,\ell}-\hat{\Phi}_{k,\ell})} + e^{-\cos(\Phi_{k,\ell}-\hat{\Phi}_{k,\ell})}\right).\end{aligned} \tag{8}$$

A problem in training DNNs with $\mathcal{L}_{\text{vmm}}(\boldsymbol{\Phi},\hat{\boldsymbol{\Phi}})$ is that the reconstructed phase may be inconsistent. Specifically, some phase elements may converge to the phase of $x(n)$ while others to the phase of $-x(n)$. To ensure a consistent reconstructed phase, the authors used the instantaneous frequency (IF) and GD losses to enhance the dependencies of phase elements along time and frequency. IF [23] is defined as the time derivative of the phase as

$$V_{k,\ell} = \mathcal{P}(\Phi_{k,\ell+1} - \Phi_{k,\ell}). \tag{9}$$

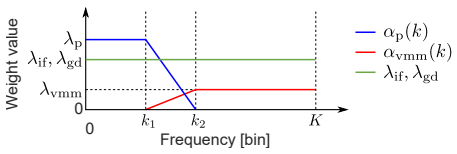


Figure 1: *Illustration of weights of* $\mathcal{L}_{\text{fw}}$.

The loss function combining IF and GD losses is defined as

$$\mathcal{L}_{\text{vmmifgd}}(\boldsymbol{\Phi},\hat{\boldsymbol{\Phi}}) = -\sum_{k,\ell}\left(\lambda_{\text{vmm}}\mathcal{C}^{\text{vmm}}_{k,\ell} + \lambda_{\text{if}}\mathcal{C}^{\text{if}}_{k,\ell} + \lambda_{\text{gd}}\mathcal{C}^{\text{gd}}_{k,\ell}\right), \tag{10}$$

where

$$\mathcal{C}^{\text{if}}_{k,\ell} = \cos(V_{k,\ell} - \hat{V}_{k,\ell}), \tag{11}$$

and λ_{vmm} and λ_{if} are the weights for the loss components.

3. Proposed phase reconstruction

In this section, we introduce several improvements for the DNN-based phase reconstruction. Specifically, we present loss functions that incorporate weights in Section 3.1 and use the extended GD in Section 3.2. In addition, we describe a data augmentation scheme for training the DNN in Section 3.3.

3.1. Weighted loss functions

In our loss functions, we combine the ideas from both [1] and [2]. For low frequencies, we use the *von Mises* distribution-based phase loss, $\mathcal{L}_{\text{p}}$, which we have found to be effective through empirical testing. At high frequencies, we utilize the *von Mises* mixture model-based phase loss, $\mathcal{L}_{\text{vmm}}$, to mitigate the sign indetermination problem. We also include the IF and GD losses to enhance the time and frequency dependencies of the phase. Weights are incorporated into the loss function as follows.

3.1.1. Frequency-specific weights

Instead of using a fixed weight for all frequency bins, we utilize weights that vary in accordance with frequencies to control the impact of $\mathcal{L}_{\text{p}}$ and $\mathcal{L}_{\text{vmm}}$ on the loss function. The loss function with frequency-specific weights is defined as

$$\begin{aligned}\mathcal{L}_{\text{fw}}(\boldsymbol{\Phi},\hat{\boldsymbol{\Phi}}) = -\sum_{k,\ell}\Big(&\alpha_{\text{p}}(k)\mathcal{C}^{\text{p}}_{k,\ell} + \alpha_{\text{vmm}}(k)\mathcal{C}^{\text{vmm}}_{k,\ell} \\ &+\lambda_{\text{if}}\mathcal{C}^{\text{if}}_{k,\ell} + \lambda_{\text{gd}}\mathcal{C}^{\text{gd}}_{k,\ell}\Big),\end{aligned} \tag{12}$$

where $\alpha_{\text{p}}(k)$ and $\alpha_{\text{vmm}}(k)$ are the weights of $\mathcal{C}^{\text{p}}_{k,\ell}$ and $\mathcal{C}^{\text{vmm}}_{k,\ell}$, respectively. Our preliminary idea for the weights is illustrated in Fig. 1, in which $\mathcal{C}^{\text{p}}_{k,\ell}$ is used at low frequencies with high weights while $\mathcal{C}^{\text{vmm}}_{k,\ell}$ is assigned low weights to reduce the effect of unstable phase elements at high frequencies. k_1 and k_2 are the boundary frequencies for the phase losses. The weights for $\mathcal{C}^{\text{if}}_{k,\ell}$ and $\mathcal{C}^{\text{gd}}_{k,\ell}$ are constant because, unlike the phase, the values of the IF and GD are not linearly dependent on frequency.

3.1.2. Amplitude weights

In addition to frequency-specific weights, we introduce amplitude weights to emphasize the importance of high-amplitude TF bins. By incorporating amplitude weights, (12) becomes

$$\begin{aligned}\mathcal{L}_{\text{afw}}(\boldsymbol{\Phi},\hat{\boldsymbol{\Phi}}) = -\sum_{k,\ell}W_{k,\ell}\Big(&\alpha_{\text{p}}(k)\mathcal{C}^{\text{p}}_{k,\ell} + \alpha_{\text{vmm}}(k)\mathcal{C}^{\text{vmm}}_{k,\ell} \\ &+\lambda_{\text{if}}\mathcal{C}^{\text{if}}_{k,\ell} + \lambda_{\text{gd}}\mathcal{C}^{\text{gd}}_{k,\ell}\Big),\end{aligned} \tag{13}$$

where

$$W_{k,\ell} = \begin{cases} |X_{k,\ell}|, & \text{if } |X_{k,\ell}| < \gamma \\ \gamma, & \text{otherwise} \end{cases}, \tag{14}$$

and γ is the weight cutoff, which is used to reduce the gap between low and high amplitudes, thereby preventing the model from excessively fitting the phase at high-amplitude TF bins.

3.2. Extension of group delay

Conventional loss functions utilize the GD loss to preserve the phase relationship across frequencies. However, as a phase difference between two consecutive bins as in (4), the GD may only capture the local relationship. Meanwhile, all frequency bins in a frame are interdependent because they are calculated from all the samples in the signal frame. To enhance the connections of the reconstructed phase elements along the frequency, we extend the calculation of the GD to the phase difference between two frequency bins with the frequency hop of i bins. The extended GD is calculated as

$$U_{k,\ell}^{(i)} = \mathcal{P}(\Phi_{k,\ell} - \Phi_{k+i,\ell}). \tag{15}$$

For $i = 1$, $U_{k,\ell}^{(i)}$ is identical to $U_{k,\ell}$. As a phase difference, $U_{k,\ell}^{(i)}$ is also a circular variable, which can be modeled by *von Mises* distribution. We integrate $U_{k,\ell}^{(i)}$ into the loss function as

$$\begin{aligned} \mathcal{L}_{\text{afw\_gd+}}(\boldsymbol{\Phi}, \hat{\boldsymbol{\Phi}}) = -\sum_{k,\ell} W_{k,\ell} \Big(& \alpha_{\text{p}}(k)\mathcal{C}_{k,\ell}^{\text{p}} + \alpha_{\text{vmm}}(k)\mathcal{C}_{k,\ell}^{\text{vmm}} \\ & + \lambda_{\text{if}}\mathcal{C}_{k,\ell}^{\text{if}} + \sum_{i \in \boldsymbol{S}} \lambda_{\text{gd}(i)}\mathcal{C}_{k,\ell}^{\text{gd}(i)} \Big), \end{aligned} \tag{16}$$

where $\boldsymbol{S}$ is a set of frequency hops used to calculate $U_{k,\ell}^{(i)}$. $\mathcal{C}_{k,\ell}^{\text{gd}(i)}$ is defined similarly to $\mathcal{C}_{k,\ell}^{\text{gd}}$, and $\lambda_{\text{gd}(i)}$ is its weight.

It is worth noting that the same technique can be applied to the IF to enhance phase relationships along the time. For the scope of this paper, we only consider the GD extension.

3.3. Data augmentation

The phase is well-known to be sensitive to the waveform shift, as even a small shift of the signal in the time domain can lead to a significant change in the phase spectrogram. Fig. 2 illustrates the amplitude and phase spectra of two signal frames with the shift of 1 sample. We can see here that the phase spectra are very different while the amplitude spectra are almost the same. When training DNNs to estimate the spectral information, conventional methods usually calculate the STFT of the signal once and use it for every epoch. In other words, since the typical window shift is larger than 1 sample, if one frame in Fig. 2 is used for training, the other will be ignored, even though both frames contain useful information about variations of the phase.

To augment the training data, for each epoch, we randomly shift the signals by m samples before calculating the STFT. The shifted signal is defined as

$$x'(n) = x(n + m). \tag{17}$$

This is equivalent to removing the first m samples of the signal. The shift m is limited in $[0, R)$. If m is equal to the window shift R, frame ℓ of $x'(n)$ is identical to frame $(\ell + 1)$ of $x(n)$.

The augmentation can be extended in cases where high-resolution data are available. By shifting the signal before resampling it to the target sampling rate, we will be able to achieve a shift of less than 1 sample.

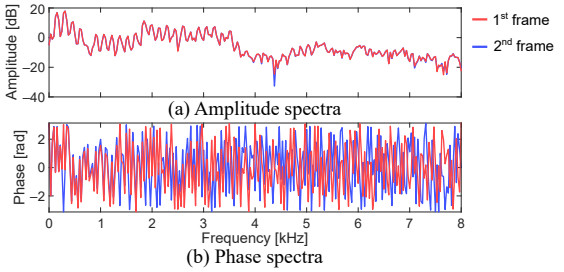


Figure 2: *Example of two frames with shift of* 1 *sample.*

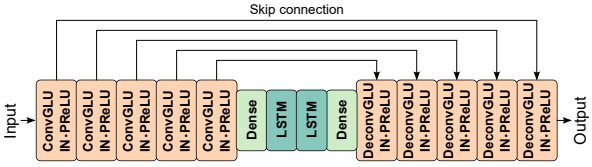


Figure 3: *Diagram of convolutional recurrent network.*

4. Experiments and results

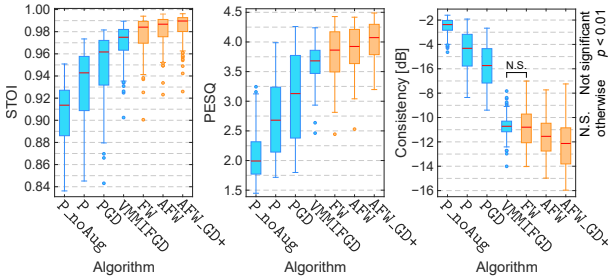
4.1. Experimental setup

The experiment is divided into two parts. First, we compared the performances of the DNN-based phase reconstruction using the proposed loss functions $\mathcal{L}_{\text{fw}}$ (hereafter, `FW`), $\mathcal{L}_{\text{afw}}$ (hereafter, `AFW`), and $\mathcal{L}_{\text{afw\_gd+}}$ (hereafter, `AFW_GD+`) with the conventional loss functions $\mathcal{L}_{\text{p}}$ [1] (hereafter, `P`), $\mathcal{L}_{\text{pgd}}$ [1] (hereafter, `PGD`), and $\mathcal{L}_{\text{vmmifgd}}$ [2] (hereafter, `VMMIFGD`). For a fair comparison, the proposed data augmentation was applied to all methods. To evaluate the efficacy of the data augmentation scheme, we trained a model using $\mathcal{L}_{\text{p}}$ without augmenting the data (hereafter, `P_noAug`). In the second part of the experiment, we compared the proposed method, `AFW_GD+`, with other non-DNN real-time phase reconstruction methods including RTISI (hereafter, `RTISI`) [19], SPSI (hereafter, `SPSI`) [20], and RTPGHI (hereafter, `RTPGHI`) [21]. For these non-DNN methods, we set the number of look-ahead frames to zero so that they are all causal. We also included the offline version of `RTPGHI` (i.e., `PGHI` [17]) for comparison.

The training data were the training set of the TIMIT Acoustic-Phonetic Continuous Speech Corpus [24], which consists of recordings of 462 speakers of eight dialects of American English each reading ten sentences. The sampling rate is 16 kHz. The test was conducted on 300 samples (150 men and 150 women) randomly selected from the test set of TIMIT.

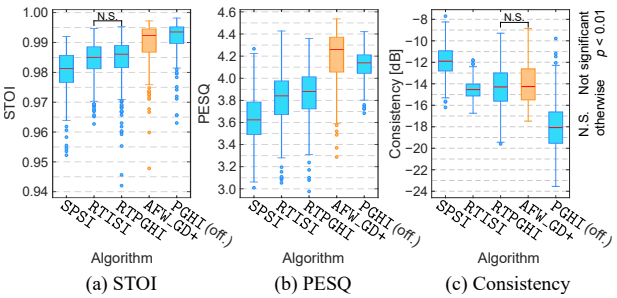
The weights were selected empirically for this preliminary work. We fixed λ_{if} and λ_{gd} to 1.0 and then varied the other weights for several values around 1.0. As a result, for all loss functions, λ_{p} was set to 1.0, and λ_{vmm} was set to 0.1. The boundary frequencies k_1 and k_2 were set to 25 and 100, respectively. After normalizing the speech signals to the active level [25] of -30 dB, the cutoff γ was set to 0.07. For $\mathcal{L}_{\text{afw\_gd+}}$, we considered only one extension of the GD, i.e., $\boldsymbol{S} = \{1, 2\}$, corresponding to the weights $\lambda_{\text{gd}(1)} = 1.0$ and $\lambda_{\text{gd}(2)} = 0.1$.

For real-time phase reconstruction, we utilized a causal DNN architecture, i.e., the convolutional recurrent network (CRN) [26], as shown in in Fig. 3. The encoder and decoder were designed symmetrically, each comprised five convolutional/deconvolutional layers with gated linear units [27] (ConvGLU/DeconvGLU). For each layer, we used a kernel size of 2×3 (time $\times$ frequency), stride of $(1, 2)$, and number of channels of 64. We also applied the instance normalization (IN) [28]



(a) STOI (b) PESQ (c) Consistency

Figure 4: *Performances of different loss functions for DNN-based phase reconstruction, where blue and red respectively indicate conventional and proposed methods.*



(a) STOI (b) PESQ (c) Consistency

Figure 5: *Performances of real-time phase reconstruction algorithms (except PGHI).*

and parametric rectified linear unit (PReLU) after each layer, except for the last layer. Temporal information was modeled by two layers of long short-term memory (LSTM) with 256 units per layer. The dense layers were utilized to convert the dimensions of the output/input of the encoder/decoder to the input/output of the LSTM layers. In total, the model consisted of nearly 2.2 million parameters.

The input of the models was the log amplitude spectrogram normalized to zero mean and unit variance. The output was the phase spectrogram. The STFT was calculated using a Hamming window with a 32-ms length, 8-ms shift, and 512-point DFT. The Adam optimizer was used with a batch size of 4 audio samples and the learning rate of 10^{-5}. For the first part of the experiments, each model was trained for 1000 epochs. The model in the second part was trained for 10 000 epochs.

As evaluation metrics, we calculated the short-time objective intelligibility (STOI) [29] and the perceptual evaluation of speech quality (PESQ) [30] of the reconstructed signals. We also calculated the consistency measure [14] as

$$C(\hat{\boldsymbol{X}}) = 10\log_{10}\frac{\|\hat{\boldsymbol{X}} - \mathrm{STFT}(\mathrm{ISTFT}(\hat{\boldsymbol{X}}))\|^2}{\|\hat{\boldsymbol{X}}\|^2}, \qquad (18)$$

where $(\hat{\boldsymbol{X}})_{k,\ell} = |X_{k,\ell}|e^{j\hat{\Phi}_{k,\ell}}$. PESQ and STOI are expected to be high, while the consistency measure is expected to be low.

4.2. Experimental results and discussion

Fig. 4 compares the performances of loss functions for DNN-based phase reconstruction, which exhibit a similar pattern for all metrics. In the comparison of the first two methods, `P` performs notably better than `P_noAug`, even though they use the same loss function $\mathcal{L}_{\mathrm{p}}$. This highlights the efficacy of the proposed data augmentation. Although the shifting technique is not a novel approach, it may have been overlooked in training DNNs for estimating the amplitude because the amplitude changes slowly over time, and this type of data augmentation may not have much of an effect. However, the experimental results here demonstrate that the shifting technique can be highly beneficial for phase reconstruction. Fig. 4 also shows that, in comparison with the conventional loss functions, the proposed loss functions `FW`, `AFW`, and `AFW_GD+` gradually lead to a better performance, thereby demonstrating the efficacy of the frequency-specific weights, amplitude weights, and the extended GD in DNN-based phase reconstruction. We have found that these DNN-based models for phase reconstruction perform better when the fundamental frequency of the signal is low and become less stable when the fundamental frequency is high. This leads to the large ranges of their scores as well as overlaps between these score distributions. However, the paired sample t-test demonstrated that all the improvements are statistically significant, except between the consistency measures of `VMMIFGD` and `FW`. In addition, the final proposed method, `AFW_GD+`, clearly outperforms all conventional methods.

Fig. 5 presents a comparison of real-time phase reconstruction algorithms, with the offline algorithm `PGHI` as a reference. The results here reveal that the proposed method achieves superior performances in PESQ and STOI while maintaining a comparable consistency measure to other real-time algorithms. In addition, the proposed method even outperforms the offline `PGHI` algorithm in PESQ and slightly underperforms in STOI. These results demonstrate the efficacy of the DNN-based method in real-time phase reconstruction.

Another advantage of the DNN-based methods is their flexibility for adaptation to various applications. For example, when a noisy/mixed phase is available, it can easily be incorporated as input to improve the performance of the model. A drawback of the conventional non-DNN methods is that they usually require a clean amplitude to estimate the phase. In contrast, DNNs can estimate the phase by using any features that contain the phase information, even if they are not clean. However, the DNN-based phase reconstruction still faces the challenge of rapid phase changes at high frequencies. Although this paper proposes using low weights for the phase loss to reduce the sensitivity, it does not fully address the problem of effectively reconstructing the high-frequency phase. Possible directions for future work include utilizing other advanced DNN architectures to better model the phase sensitivity and incorporating other phase features to enhance the phase structure.

5. Conclusions

In this paper, we presented improvements to DNN-based real-time phase reconstruction. Utilizing the varying properties of the phase as a basis, we proposed loss functions that incorporate frequency-specific weights, amplitude weights, and an extension of the GD. In addition, we introduced a data augmentation method to improve the model generalization. Experimental results demonstrated the efficacy of the data augmentation and the superior performance of the proposed loss functions compared to conventional loss functions. The results also showed that the proposed method outperforms other non-DNN real-time phase reconstruction methods.

6. Acknowledgements

This work was partly supported by the Ritsumeikan Advanced Research Academy (RARA), the Ritsumeikan Global Innovation Research Organization (R-GIRO), and by JSPS KAKENHI Grant Nos. JP20K19827, JP19H04142, JP21H03488, JP21H04427, and JP21K18372.

7. References

[1] S. Takamichi, Y. Saito, N. Takamune, D. Kitamura, and H. Saruwatari, "Phase reconstruction from amplitude spectrograms based on von-Mises-distribution deep neural network," in *2018 International Workshop on Acoustic Signal Enhancement (IWAENC)*, Sept. 2018, pp. 286–290.

[2] B. T. Nguyen, Y. Wakabayashi, G. Yuting, K. Iwai, and T. Nishiura, "Von mises mixture model-based DNN for sign indetermination problem in phase reconstruction," in *2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Nov. 2022, pp. 957–961.

[3] N. Takahashi, P. Agrawal, N. Goswami, and Y. Mitsufuji, "Phasenet: Discretized phase modeling with deep neural networks for audio source separation," in *INTERSPEECH*, Sept. 2018, pp. 2713–2717.

[4] J. Le Roux, G. Wichern, S. Watanabe, A. Sarroff, and J. R. Hershey, "Phasebook and friends: Leveraging discrete representations for source separation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 370–382, Mar. 2019.

[5] Y. Masuyama, K. Yatabe, Y. Koizumi, Y. Oikawa, and N. Harada, "Phase reconstruction based on recurrent phase unwrapping with deep neural networks," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020, pp. 826–830.

[6] L. Thieling, D. Wilhelm, and P. Jax, "Recurrent phase reconstruction using estimated phase derivatives from deep neural networks," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Jun. 2021, pp. 7088–7092.

[7] B. T. Nguyen, Y. Wakabayashi, K. Iwai, and T. Nishiura, "Two-stage phase reconstruction using DNN and von Mises distribution-based maximum likelihood," in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Dec. 2021, pp. 995–999.

[8] Y. Wakabayashi, T. Fukumori, M. Nakayama, T. Nishiura, and Y. Yamashita, "Single-channel speech enhancement with phase reconstruction based on phase distortion averaging," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 26, no. 9, pp. 1559–1569, Sept. 2018.

[9] P. Mowlaee and J. Kulmer, "Harmonic phase estimation in single-channel speech enhancement using phase decomposition and SNR information," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 9, pp. 1521–1532, Sept. 2015.

[10] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, "Phase processing for single-channel speech enhancement: History and recent advances," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 55–66, Mar. 2015.

[11] Y. Wakabayashi, T. Fukumori, M. Nakayama, T. Nishiura, and Y. Yamashita, "Phase reconstruction method based on time-frequency domain harmonic structure for speech enhancement," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2017, pp. 5560–5564.

[12] P. Magron, R. Badeau, and B. David, "Model-based STFT phase recovery for audio source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 6, pp. 1095–1105, Jun. 2018.

[13] D. Griffin and J. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, Apr. 1984.

[14] J. Le Roux, H. Kameoka, N. Ono, and S. Sagayama, "Fast signal reconstruction from magnitude STFT spectrogram based on spectrogram consistency," in *International Conference on Digital Audio Effects (DAFx)*, vol. 10, Sept. 2010, pp. 397–403.

[15] Y. Masuyama, K. Yatabe, and Y. Oikawa, "Griffin–Lim like phase recovery via alternating direction method of multipliers," *IEEE Signal Processing Letters*, vol. 26, no. 1, pp. 184–188, Nov. 2018.

[16] T. Peer, S. Welker, and T. Gerkmann, "Beyond griffin-lim: Improved iterative phase retrieval for speech," in *2022 International Workshop on Acoustic Signal Enhancement (IWAENC)*, Sept. 2022, pp. 1–5.

[17] Z. Průša, P. Balazs, and P. L. Søndergaard, "A noniterative method for reconstruction of phase from STFT magnitude," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 5, pp. 1154–1164, Mar. 2017.

[18] Y. Masuyama, K. Yatabe, Y. Koizumi, Y. Oikawa, and N. Harada, "Deep Griffin–Lim iteration: Trainable iterative phase reconstruction using neural network," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 1, pp. 37–50, Oct. 2020.

[19] X. Zhu, G. T. Beauregard, and L. L. Wyse, "Real-time signal estimation from modified short-time Fourier transform magnitude spectra," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1645–1653, Jun. 2007.

[20] G. T. Beauregard, M. Harish, and L. Wyse, "Single pass spectrogram inversion," in *2015 IEEE international conference on digital signal processing (DSP)*, July. 2015, pp. 427–431.

[21] Z. Pruša and P. L. Søndergaard, "Real-time spectrogram inversion using phase gradient heap integration," in *International Conference on Digital Audio Effects (DAFx)*, Sept. 2016, pp. 17–21.

[22] R. M. Hegde, H. A. Murthy, and V. R. R. Gadde, "Significance of the modified group delay feature in speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 190–202, Jan. 2007.

[23] B. Boashash, "Estimating and interpreting the instantaneous frequency of a signal. I. Fundamentals," *Proceedings of the IEEE*, vol. 80, no. 4, pp. 520–538, Apr. 1992.

[24] J. S. Garofolo, *TIMIT acoustic phonetic continuous speech corpus*. Linguistic Data Consortium, 1993.

[25] R. ITU, "P. 56: Objective measurement of active speech level," *International Telecommunication Union, Telecommunication Standardization Sector (ITU-T)*, 2011.

[26] K. Tan and D. Wang, "Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 380–390, Nov. 2019.

[27] A. Van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves *et al.*, "Conditional image generation with PixelCNN decoders," *Advances in neural information processing systems*, vol. 29, 2016.

[28] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[29] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, Feb. 2011.

[30] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)–a new method for speech quality assessment of telephone networks and codecs," in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, May 2001, pp. 749–752.