PhaseNet: Discretized Phase Modeling with Deep Neural Networks for Audio Source Separation

Naoya Takahashi1, Purvi Agrawal2, Nabarun Goswami3, Yuki Mitsufuji1

1Sony Corporation
2Indian Institute of Science
3Sony India Software Centre

{Naoya.Takahashi, Nabarun.Goswami, Yuki.Mitsufuji}@sony.com, purvia@iisc.ac.in

Abstract

Previous research on audio source separation based on deep neural networks (DNNs) mainly focuses on estimating the magnitude spectrum of target sources and typically, phase of the mixture signal is combined with the estimated magnitude spectra in an ad-hoc way. Although recovering target phase is assumed to be important for the improvement of separation quality, it can be difficult to handle the periodic nature of the phase with the regression approach. Unwrapping phase is one way to eliminate the phase discontinuity, however, it increases the range of value along with the times of unwrapping, making it difficult for DNNs to model. To overcome this difficulty, we propose to treat the phase estimation problem as a classification problem by discretizing phase values and assigning class indices to them. Experimental results show that our classification-based approach 1) successfully recovers the phase of the target source in the discretized domain, 2) improves signal-to-distortion ratio (SDR) over the regression-based approach in both speech enhancement task and music source separation (MSS) task, and 3) outperforms state-of-the-art MSS.

Index Terms: phase modeling, quantized phase, deep neural networks

1. Introduction

Audio source separation involves recovering target signals from a mixture of signals, e.g., clean speech from noisy speech or instrument signals from music. Most of the previous works tackle these problems by estimating the magnitude spectrogram of target signals in the short-term Fourier Transform (STFT) domain. The estimation is achieved by explicit modeling of target magnitude spectrograms [1–8] or by estimating time-frequency (TF) masks [9–14]. In these works, to transform the estimates back to time domain, the phase of the mixture signal is typically used along with the estimated magnitude spectrograms or masks in an ad-hoc manner. However, recent works have shown that estimating phase also improves the perceptual quality and the separation performance [15–17].

One approach to phase estimation is to promote consistency [18, 19], where it modifies the mixture phase depending on the results of the estimated magnitude such that the modified phase satisfies consistency. Some recent works [20–22] attempted to combine Wiener filtering with consistency-based techniques. The extension of the above approach incorporating sinusoid models has shown promising results [23]. However, the consistency constrain itself is not directly designed to recover the target phase.

There are few works that attempt to recover magnitude and phase concurrently. Williamson et al. proposed a twin-head DNN to infer both real and imaginary parts of the target spectrogram [24]. Several authors attempted to construct a fully complex-valued network by updating parameters based on complex back propagation [25, 26]. However, to achieve good performance, the network needs to be constrained by sparsity. Moreover, the currently available DNN frameworks such as PyTorch and Tensorflow do not support complex back propagation, thus preventing us from using the various modules that the framework supports.

In contrast with the above ideas, we focus here on phase modeling independent of magnitude estimation. The motivation is to enhance the performance of state-of-the-art networks by directly recovering the target phase, instead of applying Wiener filtering [4–6]. Despite of success in magnitude estimation, DNN is hard to model phase by the regression approach, partially due to the periodic nature of phase. Although unwrapping phase is one way to eliminate the phase discontinuity, it increases range of value along with the times of unwrapping, making it difficult for DNNs to model. To overcome this difficulty, we propose to treat phase estimation problem as a classification problem by discretizing phase values and assigning class indices to them. All the phase indices are equally treated in the discretized domain and the posterior probabilities for each class can be efficiently estimated by DNNs. The phase discretization or quantization has been intensively studied in speech/audio codings [27, 28]. However, to the best of our knowledge, this is the first attempt to apply source separation. The contributions of this work can be summarized as follows:

1. We propose to treat target phase estimation problem as a classification problem by discretizing phase values and assigning class indices to overcome the phase discontinuity problem.

2. We propose PhaseNet, which successfully learns meaningful distributions of the discretized phase, resulting in the recovering of the target phase in the discretized domain. The key points are also illustrated in detail.

3. The evaluation shows that the proposed method consistently improves signal-to-distortion ratio (SDR) over the regression-based approach in both single channel speech enhancement (SCSE) tasks and music source separation (MSS) task. Moreover, we compared PhaseNet with other approaches including state-of-the-art methods [23] and showed the advantage in MSS.
Table 1: Effect of Wiener Filtering (WF) on magnitude estimates of DNNs for MSS on DSD100 dataset. Values denote the Mean Squared Error (MSE) with respect to oracle magnitude.

<table>
<thead>
<tr>
<th>Source</th>
<th>DNN estimate</th>
<th>WF estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vocals</td>
<td>0.444</td>
<td>0.491</td>
</tr>
</tbody>
</table>

3. Discrete Phase Modeling

We assume that the periodic nature of the phase is one of the reasons that makes it difficult to apply DNNs for phase estimation. Therefore, we address this problem by casting the phase regression problem to a classification problem. The Fig. 2 illustrates the signal flow of the proposed method. During the training time, the target phase values $\angle S$ are discretized (or quantized) and encoded to one-hot vectors $\angle S^q$, such as $(1, 0, \ldots, 0)$ for index 0, so that DNNs can handle the problem as a classification problem. The DNNs are trained to predict the posterior probability of the quantized target phase indices given the mixture phase $\angle X$ through softmax distribution.

According to the sinusoidal model [29], the phase of slowly varying sinusoids can be written as:

$$\phi(f,t) = \phi(f,t-1) + 2\pi \nu h,$$

where $\phi(f,t)$, $\nu$ and $h$ denote the phase at time frame $t$, the normalized frequency, and the hop size (in samples), respectively. Equation 1 suggests that the phase of the sinusoid varies depending on the TF bins and influenced by the frame shift of the STFT window. To mitigate this modulation effect for DNN phase estimation, we compensate the effect by subtracting $2\pi th\nu$ from each TF bin, which is denoted as de-modulation in Fig. 2, and wrap to $(-\pi, \pi]$.

The phase of mixture is dominantly affected by one of the sources if the magnitude of the source is dominant in a TF bin. If the magnitude of a target source is much higher than that of an interference, the mixture phase is most probably close to the target phase. On the other hand, if the target magnitude is at similar level or lower than the interference, the phase could be tweaked by the interference and the phase of these TF bins need to be estimated. To incorporate this characteristic property, we also feed the log magnitude ratio:

$$R = \log \left( \frac{|S|}{|X|} \right)$$

(2)

to the network by concatenating it along channel dimension. The network is trained to minimize the cross entropy loss $L$:

$$L(\theta) = -\sum \angle S^q_i \log P(\phi^q|\angle X_i, R_i, \theta),$$

(3)

where $\angle S^q_i (q = 1, \ldots, Q)$ denotes the index of one-hot encoded quantized phase, $P(\phi^q|\angle X_i, R_i, \theta)$ is the softmax output of
DNN for quantized phase $\phi$ given ith sample. The quantization level and the network parameters are denoted as $Q$ and $\theta$ respectively. During the inference time, when the magnitude of target source $|S|$ is not available, it is estimated by any method to provide the log magnitude ratio $\hat{R}$ as an input to DNN. We can also use the estimated source magnitude $|\hat{S}|$ for training or fine tuning to improve the phase estimation. The index that has the maximum probability, $\hat{q} = \arg\max_{q} P(\phi|\hat{X}_i, \hat{R}, \theta)$ is used to transform back to the quantized phase value $\phi^{\hat{q}}$. Hereafter, we call the DNN trained with this approach as PhaseNet. Recent works show that even when the data is implicitly continuous, the discrete softmax distribution works better [30, 31]. Moreover, the recent success of DNN based image classification methods suggest that converting continuous image to discrete class would not be a problem. In the discrete representation, every quantized point is treated equally and there is no explicit assumption on data, e.g., no periodic nature as Fig. 3 illustrates. However, the PhaseNet successfully learned a meaningful relationship among phase classes as discussed in Section 4.4.

4. Experiments

4.1. Quantization level

To assess the impact of the quantizing phase, we first conducted a subjective test. Speech signals from the Wall Street Journal (WSJ0) corpus were transformed into STFT, the phase was uniformly quantized by a different number of levels and was transformed back to the time domain signal. Ten audio engineers participated in the subjective test. Audio is presented with Sony’s headphone 900ST. Six sentences from 3 male and 3 female speakers and 3 quantization levels (4, 8 and 12) per sentence were prepared for the test. The subjective test was conducted in a similar way to the double-blind triple-stimulus with hidden reference format (ITU-R BS.1116), where the reference was the original speech signal and one among A and B was same as the reference, the other being the quantized phase presented in a random order. The subjects were asked to identify which one was the same as the reference signal among A and B. This resulted in 60 evaluations for each quantization level. The Fig. 4 summarizes results. In the figure, blue bars indicate the accuracy of finding the reference signal from A and B at quantization levels 4, 8 and 12. The red plot indicates the average SDR values for each quantization level. As can be observed, the accuracy of finding the correct reference signal is closer to the chance rate (50%) for quantization levels 8 and 12. From this subjective test, we interpreted that quantization level 8 and above, there is no noticeable difference from the reference signal perceptually.

4.2. Single channel speech enhancement (SCSE)

Next, we evaluated the proposed method on the single channel speech enhancement task. The dataset used for training was the speaker independent subset of the WSJ0 corpus. For noise source, the 3rd CHiME challenge (4 types noise) and AE Dataset [32, 33] (41 types noise) were used. AE Dataset was down sampled to 16kHz to match the sampling rate and the original train/test split was used. For the noise data from CHiME, we used session number 040 as a test set. The training data was prepared by randomly mixing sources of varying SNR from $-7$ to 6 dB. The STFT was performed with a frame size of 1024 samples with 75 % overlap. The PhaseNet architecture was adapted from the MDenseNet architecture proposed in [5]. Table 2 presents the details of the architecture, where $l$ denotes the number of layers and $k$ denotes growth factor of each dense block. The final layer of PhaseNet has $\#Q \times \#ch$ number of feature maps, where $\#Q$ is the number of quantization levels equal to 16 and $\#ch$ is the number of channels in the audio equal to 1. The PhaseNet was trained with Adam optimizer until the loss curve plateaued.

We consider three baselines for comparison, the lower baseline which uses mixture phase, the upper baseline which uses an oracle phase, and phase from a DNN trained with regression approach (DNN-R). The DNN architecture of DNN-R is identical to PhaseNet except the last layer where the softmax output for classification is replaced with a standard convolution output. The input of DNN-R is same as PhaseNet and it is trained to estimate the difference of the target phase and mixture phase ($\angle S - \angle X$) by minimizing the mean square error (MSE).

For reconstructing the time domain target signal, we considered two cases, namely oracle magnitude and noisy magnitude, since the magnitude of the target source is estimated by some method in inference time, and that estimate is usually not perfect. We simulated the noisy magnitude estimate by mixing the noise source in the input with $-18$ dB attenuation.

Table 2: PhaseNet architecture based on MDenseNet [5].

<table>
<thead>
<tr>
<th>scale</th>
<th>$l$</th>
<th>$\frac{1}{2}$</th>
<th>$\frac{1}{2}$</th>
<th>$\frac{5}{8}$</th>
<th>$\frac{1}{16}$</th>
<th>$\frac{1}{8}$</th>
<th>$\frac{1}{4}$</th>
<th>$\frac{1}{2}$</th>
<th>$1$</th>
<th>$#Q \times #ch$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$4$</td>
<td>$16$</td>
<td>$18$</td>
<td>$16$</td>
<td>$16$</td>
<td>$16$</td>
<td>$16$</td>
<td>$16$</td>
<td>$18$</td>
<td>$4$</td>
<td>$1$</td>
</tr>
</tbody>
</table>
3 compares signal to distortion ratios (SDRs) of estimated target signal are compared with baselines in three SNR scenario, namely -6, -3, 0 dB. The results shows that the proposed method consistently outperform lower baselines and the regression approach. As the SNR becomes low, the input phase is more likely to be dominated by noise. Even in this case, PhaseNet improve the SDR more robustly. It should be noted that even though the PhaseNet was trained only on the clean source magnitude, it significantly improved the performance even when the source magnitude was not perfect.

### 4.3. Music source separation (MSS)

In this section we describe the evaluation of the proposed method on the music source separation task. Specifically, focused on singing voice separation, where the vocals need to be extracted from a mixture of musical sources. For the evaluation we used the Demixing Secrets Database (DSD100), released as part of the SiSEC campaign [34], downsampled to a sampling rate of 22.05kHz. In DSD100, the mixture and its four sources - bass, drums, vocals, and other, are available. Thus, our task was to recover the phase of vocals $\angle S$ from the song $x$. For the MSS task, we used quantization level $\#Q = 20$. The STFT was performed with frame size of 2048 samples with 75% overlap. The PhaseNet architecture was the same as that used in the SCSE task up to the final layer, where it was changed based on the $\#Q$ and $\#ch$ values. The network was trained to estimate the quantized phase index $\angle S^q$ with the CE loss with Adam optimizer. The initial learning rate of 0.001, reduced to 0.0001 after training curve saturated. Similar to the SCSE task, to reconstruct the time domain signal, we considered two scenarios, oracle magnitude and estimated magnitude. For a realistic evaluation, we used a MM.DenseNet [5] to estimate the magnitude of the target source. In addition to the baselines mentioned in section 4.2, we compared PhaseNet with consistent anisotropic Wiener filtering (CAW), which showed superior performance to Wiener filtering, consistent Wiener filtering and anisotropic Wiener filtering [23].

The SDR values on Test set are compared in Table 4. From the results, it can be observed that the phase estimated by PhaseNet gives an absolute improvement of about 3.2dB SDR over lower baseline with oracle magnitude and 1.5dB SDR with estimated magnitude. Also worth noting is that PhaseNet performs as well as CAW with oracle magnitude, but more robustly improves performance in the realistic scenario of estimated magnitudes.

### 4.4. Estimated phase distribution

As described in Section 3, since PhaseNet is trained as a classification problem to predict quantized target phase indices, there is no assumption about the data such as periodicity and closeness of discretized points. Therefore, it is worth investigating how PhaseNet outputs are distributed. Fig. 5 shows a normalized histogram of the difference of indices $\delta q$ between the target phase and the phase inferred by PhaseNet in Section 4.3. $\delta q = 0$ indicates that the phase is correctly recovered, $\delta q = 1$ indicates that the phase is wrongly estimated to a closest neighboring point, $\delta q = 2$ indicates the estimate is the second neighbor of the target, and so on ($\delta q = 10$ indicates the estimate is the opposite phase in case $\#Q = 20$). For comparison, the histogram of the mixture-source index difference was also presented. The histograms show that PhaseNet shifted the peak of the histogram to $\delta q = 0$ and more rapidly decayed toward the opposite phase, in comparison with the mixture-source index difference. It suggests that the PhaseNet learned a natural posterior distribution that has clear peak at target phase, and was aware of “neighboring points”.

### 5. Conclusion

We proposed to treat the phase estimation problem as a classification problem, and proposed PhaseNet that can be used with any magnitude estimation method. The experimental results showed that 1) the quantizing phase at a reasonable level does not degrade the perceptual quality, 2) PhaseNet improved SDRs over the regression-based approach in SCSE tasks and 3) PhaseNet outperformed state-of-the-art in MSS task, and robustly improved SDRs even if the magnitude estimates were imperfect.
6. References


