PHASE SPECTRUM PREDICTION OF AUDIO SIGNALS

Ali Bahrami Rad *, *Tuomas Virtanen*

Department of Signal Processing, Tampere University of Technology
Korkeakoulunkatu 1, FI-33720, Tampere, Finland
ali.bahrami.rad@aalto.fi, tuomas.virtanen@tut.fi

ABSTRACT

Modeling the phases of audio signals has received significantly less attention in comparison to the modeling of magnitudes. This paper proposes to use linear least squares and neural networks to predict phases from the neighboring points only in the phase spectrum. The simulation results show that there is a structure in the phase components which could be used in further analysis algorithms based on the phase spectrum.

Index Terms— STFT, phase spectrum prediction, phase unwrapping, linear least squares, neural networks

1. INTRODUCTION

A large amount of audio signal processing and analysis algorithms operate in the time-frequency domain. Most of the analysis algorithms use the magnitudes of the time-frequency points only. There are some analysis methods which use the phase spectrum information [1, 2, 3], but majority of the analysis systems do not use the phase information at all.

Many audio processing methods operate using the magnitudes only, and then either apply the original phases [4], or find phases so that they match with the processed magnitudes [5, 6]. The main reason why phases are not commonly used in the analysis of audio signals, or why phases are not often used in the processing of sounds, is the stochastic nature of phases. In comparison to magnitudes, phases are usually significantly more difficult to model [7]. Furthermore, it has been assumed that the human auditory system is insensitive to phases [8], even though it has been shown that the phase spectrum affects the perception of sounds, at least in some situations [9, 10].

Although the phase and the magnitude of the short-time Fourier transform (STFT) are strongly related [5], this paper investigates the predictability of phases by using only phase information in the time-frequency domain. We propose to model phases by linear regression using phases in the neighboring time-frequency points. The simulation results show that there is a structure in phases, which allows predicting them, at least to some degree. The regression coefficients or the residual that our methods produce could be used as the basis for novel analysis algorithms based on the phase spectrum.

The structure of the paper is as follows: Section 2 describes the time-frequency representation of phases used in the paper. Section 3 proposed regression models for predicting phases. Sections 4, and 5 present the results and discussion.

*Currently in Department of Information and Computer Science, Aalto University, Espoo, Finland.

2. PHASE SPECTRUM

The STFT of signal $x(n)$ is denoted by $X(k, \ell N_{hop})$ where $0 \leq k \leq N_{\text{FFT}}$ is the index of the spectral line, and ℓ is the index of the time slice, and N_{hop} is the temporal decimation factor. In addition, in this work we use the Hamming window (with size N_{win}) as the analysis window. If we decompose $X(k, \ell N_{hop})$ in the following way

$$X(k, \ell N_{hop}) = |X(k, \ell N_{hop})|\, e^{\jmath \theta^{\mathscr{W}}_{k,\ell}}, \tag{1}$$

then $|X(k, \ell N_{hop})|$ and $\theta^{\mathscr{W}}_{k,\ell}$ are called the "magnitude spectrum" and the "phase spectrum" respectively. For the phase spectrum we have

$$\theta^{\mathscr{W}}_{k,\ell} = \text{atan2}\Big(X_{\Im}(k, \ell N_{hop}), X_{\Re}(k, \ell N_{hop})\Big), \tag{2}$$

where the subscripts $\Re$ and $\Im$ denote the real and imaginary parts.

From the definition of $\theta^{\mathscr{W}}_{k,\ell}$, it is obvious that $-\pi < \theta^{\mathscr{W}}_{k,\ell} \leq \pi$, thus we call it "wrapped phase" since it is wrapped around $\pm\pi$. The superscript $\mathscr{W}$ in $\theta^{\mathscr{W}}_{k,\ell}$ stands for "wrapped". Due to the discontinuity of $\theta^{\mathscr{W}}_{k,\ell}$ which makes its prediction difficult, we firstly try to unwrap it then model the unwrapped version which is a continuous function; however, this is not an easy task because by adding any multiple of 2π to $\theta^{\mathscr{W}}_{k,\ell}$ the value of $X(k, \ell N_{hop})$ does not change, thus there are an infinite number of ways to unwrap $\theta^{\mathscr{W}}_{k,\ell}$ [7].

The chosen method for phase unwrapping relies on the detection of the discontinuities between the wrapped phases at two adjacent frequencies $k-1$ and k; whenever the difference of those wrapped phases is greater than π we say that discontinuity is present [11]. For calculation of unwrapped phase spectrum $\theta_{k,\ell}$ from $\theta^{\mathscr{W}}_{k,\ell}$ we have

$$\theta_{k,\ell} = \theta^{\mathscr{W}}_{k,\ell} + \vartheta_{k,\ell} \times 2\pi \tag{3}$$

where $\vartheta_{k,\ell} \in \mathbb{Z}$ corresponds to the number of rotations in the trigonometrical circle, and can be calculated in the following way

$$\vartheta_{k,\ell} = \begin{cases} 0 & \text{for } k{=}0 \\ \vartheta_{k-1,\ell} & \text{for } k{\geq}1 \;\&\; -\pi{<}\theta^{\mathscr{W}}_{k,\ell}{-}\theta^{\mathscr{W}}_{k-1,\ell}{\leq}\pi \\ \vartheta_{k-1,\ell}+1 & \text{for } k{\geq}1 \;\&\; \theta^{\mathscr{W}}_{k,\ell}{-}\theta^{\mathscr{W}}_{k-1,\ell}{\leq}{-}\pi \\ \vartheta_{k-1,\ell}-1 & \text{for } k{\geq}1 \;\&\; \pi{<}\theta^{\mathscr{W}}_{k,\ell}{-}\theta^{\mathscr{W}}_{k-1,\ell} \end{cases}. \tag{4}$$

This method is implemented in MATLAB function unwrap(·). We use this method by a modification when the FFT size and the analysis window length are equal.

As demonstrated in Fig. 1, the unwrapped phase spectrum is dependent on the FFT size (N_{FFT}). In this paper we assume FFT size is the multiplication of a power of two by the analysis window length

$$N_{\text{FFT}} = 2^p \times N_{\text{win}}, \tag{5}$$

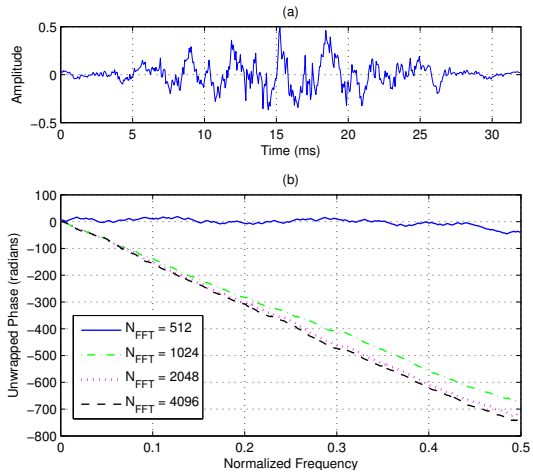

Fig. 1. (a) is the 512 samples of a short-time section of an audio signal ($N_{\text{win}} = 512$) which is 32 ms long (sampling frequency is 16 kHz). (b) represents the unwrapped phase spectra with different FFT length ($N_{\text{FFT}} = 512,\ 1024,\ 2048,$ and 4096). As can be seen the unwrapped phase spectrum is dependent on the FFT length.

where p is a non-negative integer. If we assume there is a true unwrapped phase spectrum, each of the plots in Fig. 1 is an approximation of that true unwrapped phase. But, it is obvious that this approximation is not always accurate. For instance, when FFT size is 512, we have a poor approximation of the true unwrapped phase. According to our experiments, this phenomenon happen whenever the FFT size is equal to the analysis window length ($p = 0$), and in other cases where $p \in \mathbb{N}$ we do not have this problem.

To address this problem we can use more accurate phase unwrapping methods. But, in this paper we only want to investigate the predictability of phases in phase spectrum not to propose an efficient phase unwrapping method. So, we simply modify the former phase unwrapping method such that it satisfies our needs for modeling which is to create a smooth function for the phase spectrum, and postpone the task of finding the efficient phase unwrapping method for the future work. One way to solve the problem when $p = 0$ is to calculate the unwrapped phase for $p \geq 1$ and then resample the result. But, the proposed method here which is computationally less expensive is obtained by analyzing the plots of the unwrapped phases and finding an approximation for those plots by modifying Eq. (4).

In Fig. 1 by looking at the unwrapped phases of $N_{\text{FFT}} = 1024,\ 2048,$ and 4096 we see that those graphs are the graphs of almost monotonically decreasing functions. So, if we modify the definition of $\vartheta_{k,\ell}$ (only when $p = 0$) in the following way, the resulting unwrapped phase is a monotonically decreasing function

$$\vartheta_{k,\ell} = \begin{cases} 0 & \text{for } k{=}0 \\ \vartheta_{k-1,\ell} & \text{for } k{\geq}1 \ \&\ \theta''_{k,\ell} {\leq} \theta''_{k-1,\ell} \\ \vartheta_{k-1,\ell} - 1 & \text{for } k{\geq}1 \ \&\ \theta''_{k,\ell} {>} \theta''_{k-1,\ell} \end{cases}. \tag{6}$$

Although the assumption that the unwrapped phase is a monotonically decreasing function is not true, the resulting unwrapped phase matches the unwrapped phase at higher frequency resolution (higer p) very well. As demonstrated in Fig. 2 the resulting unwrapped phase in the case of $N_{\text{FFT}} = 512$ using this method, matches the unmodified phase unwrapping method with $N_{\text{FFT}} = 4096$ which was a good approximation of the unwrapped phase.

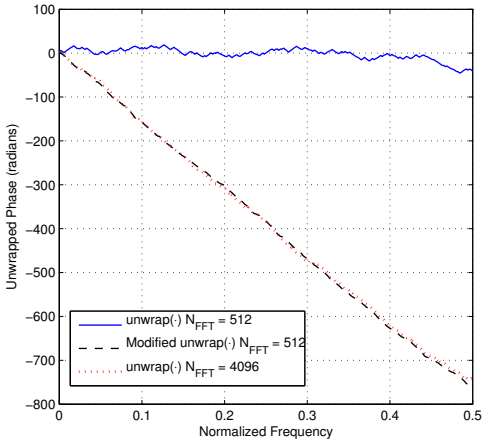

Fig. 2. The unwrapped phase spectra of the same signal as Fig. 1 with different methods.

3. PHASE SPECTRUM PREDICTION

In the previous section we produce almost smooth curves for phases in each time slice in the phase spectrum (phase unwrapping is done on the frequency axis). In this section we propose models to estimate the phase at the specific point in the phase spectrum by using the actual observed values of the phases at the neighboring points. We should consider that all the following models use neighboring points in the phase spectrum with lower frequency and frame indices (causal models). Moreover, the proposed models are not the only models to estimate the phases, and one can estimate phases by using higher order (higher number of neighboring points) or even noncausal (the neighboring points with higher frequency and frame indices) models. In this paper we only want to show the possibility to predict phases by using phase information available in the phase spectrum of STFT of signal.

If we consider Fig. 3 as a part of an unwrapped phase spectrum, then one possible estimation for the unwrapped phase at frequency k and time slice ℓ is

$$\hat{\theta}_{k,\ell} = \theta_{k-1,\ell} + \theta_{k,\ell-1} - \theta_{k-1,\ell-1}. \tag{7}$$

We are inspired to use this predictor based on the assumption that the value of $\theta_{k,\ell}$ is close to the value of $\theta_{k-1,\ell}$,

$$\theta_{k,\ell} = \theta_{k-1,\ell} + \delta_{k,\ell}, \tag{8}$$

where $\delta_{k,\ell}$ is the difference between $\theta_{k,\ell}$ and $\theta_{k-1,\ell}$. We assume that $\delta_{k,\ell}$ can be estimated by the difference between unwrapped phases of the previous time slice

$$\hat{\delta}_{k,\ell} = \theta_{k,\ell-1} - \theta_{k-1,\ell-1}. \tag{9}$$

As we will see later, this model is not very accurate, but for starting point it is a good guess. We call it the *basic model* (*BM*). In the following sections, we use the systematic ways to introduce models based on the linear least squers and neural networks.

3.1. Phase Spectrum Prediction Using Linear Least Squares Regression

In this section we introduce four different models based on the values of adjacent unwrapped phases to $\theta_{k,\ell}$ in the unwrapped phase

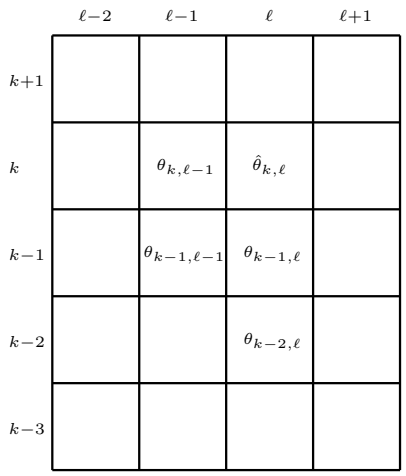

Fig. 3. A part of an unwrapped phase spectrum. Rows correspond to the spectral lines and columns correspond to the time slices. The unwrapped phase at frequency k and time slice ℓ is estimated by the adjacent unwrapped phases.

spectrum. The four predictors are given as

$$\hat{\theta}^q_{k,\ell} = \begin{cases} \lambda_0+\lambda_1\theta_{k-1,\ell} & \text{for } q=1 \\ \lambda_0+\lambda_1\theta_{k-1,\ell}+\lambda_2\theta_{k-2,\ell} & \text{for } q=2 \\ \lambda_0+\lambda_1\theta_{k-1,\ell}+\lambda_2\theta_{k,\ell-1}+\lambda_3\theta_{k-1,\ell-1} & \text{for } q=3 \\ \lambda_0+\lambda_1\theta_{k-1,\ell}+\lambda_2\theta_{k-2,\ell}+\lambda_3\theta_{k,\ell-1}+\lambda_4\theta_{k-1,\ell-1} & \text{for } q=4 \end{cases}, \tag{10}$$

where $\hat{\theta}^q_{k,\ell}$ is the predicted value for $\theta_{k,\ell}$ by using the qth model. And, λ_j are $q+1$ adjustable parameters of the qth model. Fig. 3 illustrates the structure of the predicted phase and its neighboring phases. We define vectors

$$\boldsymbol{\lambda}_q^T = \begin{cases} [\lambda_0, \lambda_1] & \text{for } q=1 \\ [\lambda_0, \lambda_1, \lambda_2] & \text{for } q=2 \\ [\lambda_0, \lambda_1, \lambda_2, \lambda_3] & \text{for } q=3 \\ [\lambda_0, \lambda_1, \lambda_2, \lambda_3, \lambda_4] & \text{for } q=4 \end{cases}, \tag{11}$$

and

$$(\boldsymbol{\psi}_q^{k,\ell})^T = \begin{cases} [1, \theta_{k-1,\ell}] & \text{for } q=1 \\ [1, \theta_{k-1,\ell}, \theta_{k-2,\ell}] & \text{for } q=2 \\ [1, \theta_{k-1,\ell}, \theta_{k,\ell-1}, \theta_{k-1,\ell-1}] & \text{for } q=3 \\ [1, \theta_{k-1,\ell}, \theta_{k-2,\ell}, \theta_{k,\ell-1}, \theta_{k-1,\ell-1}] & \text{for } q=4 \end{cases}. \tag{12}$$

Then, the vector form of the predictors are given as

$$\hat{\theta}^q = (\boldsymbol{\psi}_q)^T \boldsymbol{\lambda}_q, \tag{13}$$

where k (index of the spectral line) and ℓ (index of the time slice) are omitted for convenience. The goal is to estimate the vector of parameters $\boldsymbol{\lambda}_q$ to minimize the residual sum of squares (RSS)

$$\text{RSS}(\boldsymbol{\lambda}_q) = \sum_{i=1}^{N} (\theta_{\langle i \rangle} - \hat{\theta}^q_{\langle i \rangle})^2, \tag{14}$$

where $\langle i \rangle$ is the index of the training data, and N is the total number of training data. The vector form of Eq. (14) is

$$\text{RSS}(\boldsymbol{\lambda}_q) = (\boldsymbol{\theta} - \boldsymbol{\Psi}_q\boldsymbol{\lambda}_q)^T(\boldsymbol{\theta} - \boldsymbol{\Psi}_q\boldsymbol{\lambda}_q), \tag{15}$$

where $\boldsymbol{\Psi}_q$ is the $N \times (q+1)$ matrix of inputs and $\boldsymbol{\theta}$ is the N-vector of outputs in the training set. The solution minimizing this criterion is

$$\hat{\boldsymbol{\lambda}}_q = (\boldsymbol{\Psi}_q^T\boldsymbol{\Psi}_q)^{-1}\boldsymbol{\Psi}_q^T\boldsymbol{\theta}. \tag{16}$$

And, the vector of the estimated values for the unwrapped phases is

$$\begin{aligned} \hat{\boldsymbol{\theta}} &= \boldsymbol{\Psi}_q\hat{\boldsymbol{\lambda}}_q \\ &= \boldsymbol{\Psi}_q(\boldsymbol{\Psi}_q^T\boldsymbol{\Psi}_q)^{-1}\boldsymbol{\Psi}_q^T\boldsymbol{\theta}. \end{aligned} \tag{17}$$

In this paper the qth model using linear least squares is called *LS-q*. For example, *LS-2* means the 2nd model using linear least squares indicated by Eq. (10) when $q = 2$. In addition, it is obvious that *BM* which was described previously is a special case of *LS-3* with fixed parameters $\lambda_0 = 0$, $\lambda_1 = \lambda_2 = 1$, and $\lambda_3 = -1$.

3.2. Phase Spectrum Prediction Using Neural Networks

So far, all models *BM*, *LS-1*, *LS-2*, *LS-3*, and *LS-4* were linear functions of input variables. In this section by using neural networks, we introduce new models which are non-linear functions of inputs. These models are counterpart of the previous models, i.e. we use the same input/output structure as before in the input/output layer of the neural networks. In addition, all models use one hidden layer with 3 neurons. So, the general architecture of NN is R-3-1 in which R $\in \{1, 2, 3, 4\}$ is the number of units in the input layers, and we only have one unit in the output layer. Again, the model with R units in the input layers is called *NN*-R. For example, *NN-4* means the model with 4 units in the input layers using neural networks, and it is the counterpart of *LS-4*.

For the hidden units we use "tanh" activation function, and for the output unit we use identity activation function. Finally, for the network training the BFGS quasi-Newton algorithm [12] is used.

4. SIMULATIONS AND RESULTS

In order to test the accuracy of the proposed methods, simulation experiments were carried out.

4.1. Acoustic Material

The training data consists of approximately 15 seconds of audio signals including two segments of two different music pieces, and three different speech signals from one female and two male speakers. The test data consists of 100 different segments (in total approximately 300 seconds) of music and speech signals. The size of the training data is chosen to be smaller than the test data to speed up the training. In addition, the training and test data sets are disjoint, so the samples which are used in training are not used in testing. The music pieces are selected from the music database described in [13]. We choose music data with different genre including pop, rock, blues, metal, hip hop, and so on. The music data is re-sampled to 16 kHz. For the speech data we use the TIMIT database [14] where the sampling frequency is 16 kHz.

Table 1. Results of the phase spectrum prediction using Linear Least Squares when $N_{\text{FFT}} = N_{\text{win}}$, and $N_{\text{overlap}} = 0.5 \times N_{\text{win}}$.

	$N_{\text{FFT}} = N_{\text{win}} = 256$			$N_{\text{FFT}} = N_{\text{win}} = 512$			$N_{\text{FFT}} = N_{\text{win}} = 1024$		
	Var	SNR	Ent	Var	SNR	Ent	Var	SNR	Ent
Random	3.333	-1.50	2.65	3.289	-2.00	2.65	3.288	-2.22	2.65
BM	2.336	3.93	2.11	2.524	2.59	2.58	2.784	-0.51	2.62
LS-1	1.659	6.54	2.30	1.716	7.00	2.34	1.813	5.45	2.40
LS-2	1.622	6.84	2.28	1.656	7.47	2.31	1.675	6.77	2.34
LS-3	1.638	6.58	2.29	1.687	7.04	2.33	1.744	5.53	2.37
LS-4	1.607	6.86	2.27	1.638	7.49	2.30	1.637	6.74	2.32

4.2. Performance Evaluation of Models

If the predicted value of unwrapped phase $\theta_{k,\ell}$ is $\hat{\theta}_{k,\ell}$, then the prediction error is

$$\epsilon = \theta - \hat{\theta}, \tag{18}$$

where indices k and ℓ are omitted for convenience. We can rewrite this prediction error in the following way

$$\epsilon = \upsilon + 2\pi \times m, \tag{19}$$

where $\upsilon \in (-\pi, \pi]$ and $m \in \mathbb{Z}$. It is obvious that for different values of m, we will have the same result if a signal is to be reconstructed using the predicted phase. Thus, this error is not a good criterion to evaluate the success in prediction; instead we can use υ which is the wrapped version of ϵ. From now, whenever we say prediction error we mean υ.

The first criterion which is used for performance evaluation of different models is the variance of the prediction error (Var) of the test data set. The smaller Var means the better prediction.

The second criterion used to evaluate the performance is the entropy of the prediction error (Ent) of the test data set. Entropy is a measure of the uncertainty of a random variable [15]. The prediction error is a continuous random variable since υ can be any value in the interval $(-\pi, \pi]$. Thus, we cannot use discrete entropy; instead we use the differential entropy. The discrete entropy is positive and it is used as a measure of uncertainty. But, unlike the discrete entropy, the differential entropy is not in general a good measure of uncertainty or information. For example, the differential entropy can be any value from $-\infty$ to ∞, and it is used to measure only changes in uncertainty [16]. In other words, differential entropy does not provide an absolute measure of randomness or code length; instead it provides a relative measure of these properties [17]. In spite of this drawback, still we can use the differential entropy as a criterion for performance evaluation of the methods. A random variable Y is less predictable than Z whenever $\text{Ent}(Y) > \text{Ent}(Z)$, and an event from Y requires more bits on average to encode than an event from Z [17].

The third and the last criterion is the power signal-to-noise ratio (SNR) between the original signal $x(n)$ and the reconstructed signal $\hat{x}(n)$. To construct $\hat{x}(n)$, we take the inverse STFT of complex spectra $\hat{X}_{k,\ell}$ which is obtained from the original magnitude and the predicted phases, then followed by windowing and overlap-add (OLA) synthesis method.

Both Var and Ent are good measures to show the success in prediction of phase, but since they do not involve the magnitude information of audio signals, their relation to the quality of predicted signal are not clear. On the other hand, since SNR uses both magnitude and phase information in signal reconstruction, it is the most important criterion in evaluation of the methods and it has a direct relation to the quality of the predicted signal.

Table 2. Results of the phase spectrum prediction using neural networks when $N_{\text{FFT}} = N_{\text{win}} = 512$, and $N_{\text{overlap}} = 0.5 \times N_{\text{win}}$.

	$N_{\text{FFT}} = N_{\text{win}} = 512$		
	Var	SNR	Ent
NN-1	1.707	7.13	2.33
NN-2	1.656	7.51	2.31
NN-3	1.681	7.09	2.33
NN-4	1.633	7.52	2.30

4.3. Results of Phase Prediction

The results of the phase prediction by using linear least squares in addition to the results of the basic model and random guess are listed in Table 1 (when $N_{\text{FFT}} = N_{\text{win}} = 256$, 512, and 1024). Moreover, the results of the phase prediction by using neural networks when $N_{\text{FFT}} = N_{\text{win}} = 512$ and $N_{\text{overlap}} = 0.5 \times N_{\text{win}}$ are reported in Table 2.

5. DISCUSSION

As can be seen in Table 1, the best results belong to *LS-2* and *LS-4*; they have the lowest Var, the lowest Ent, and the highest SNR. This happen independent from FFT size. And, by increasing the FFT size still these two models are better than the others. For example, if $N_{\text{FFT}} = 4096$, $N_{\text{win}} = 512$, and $N_{\text{overlap}} = 0.5 \times N_{\text{win}}$, the SNR for *LS-2*, and *LS-4* are respectively 35.5 and 35.8 dB. But those numbers for *LS-1* and *LS-3* are 27.9 and 28.3 dB.

The common property between *LS-2* and *LS-4* is that in both models we use two unwrapped phases in the current time slice (*LS-4* uses two extra unwrapped phases from the previous time slice). Thus, in phase estimation the unwrapped phases in the current time slice have more effect than the unwrapped phases of the previous time slice (it is good to remind that the unwrapping is done in the frequency axis).

Another point which we mention here is that by increasing frequency resolution (increasing p in Eq. (5)), the performance of the proposed models will increase. For example, if $N_{\text{win}} = 512$ and $N_{\text{overlap}} = 0.5 \times N_{\text{win}}$, for $p = 0$, 1, 2, and 3 the SNR of *LS-2* are respectively 7.47, 11.77, 26.0, and 35.5 dB.

In addition, for the results in Table 1 we use modified unwrap$(\cdot)$ function which was described in Section 2. But, if we use the original MATLAB unwrap$(\cdot)$ function, the SNR (when $N_{\text{FFT}} = N_{\text{win}} = 512$) for *LS-1* to *LS-4* will be -5.1, -4.8, -5.1, and -4.8 dB. This shows the crucial role of phase unwrapping method in the phase modeling task.

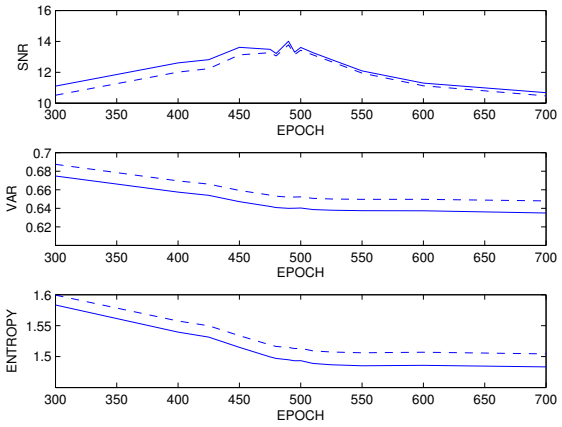

Fig. 4. The results of training the neural network for different EPOCH count when $N_{\text{FFT}} = 1024$, $N_{\text{win}} = 512$, and $N_{\text{overlap}} = 0.5 \times N_{\text{win}}$. The dashed lines are the results of the test data. The other lines are the results of the training data.

As can be seen from Table 2 the result of neural networks are slightly better than linear least squares. Basically, we expect to get the better result by using neural networks than linear least squares. But, it is not easy, because the objective function in training the neural networks is the mean-square error of unwrapped phases. So, by progress in training process the value of mean-square error decreases, but it does not mean that the result will have the higher SNR. Thus, it would be better to change the objective function of the neural network to some criterion which is related to SNR (we postpone this for future works). The results of training the neural network for different number of epochs are plotted in Fig 4.

And finally, because the number of parameters in the linear least squares and also the number of neurons in the neural network are not large we do not need to concern about overfitting problem.

6. CONCLUSIONS

In this paper, different methods for phase prediction from only phase information have been proposed. The methods are based on the linear least squares and neural networks. The simulation results show that there exists a structure in the phase spectrum that allows predicting the phase using the neighboring phases at least to some degree.

7. REFERENCES

[1] L. D. Alsteris, and K.K. Paliwal,, "ASR on speech reconstructed from short-time Fourier phase spectra," in *Proc. International Conf. Spoken Language Processing*, October 2004.

[2] F. J. Charpentier, "Pitch detection using the short-term phase spectrum," in *Proc. IEEE International Conf. on Acoustics, Speech, and Signal Processing*, pp. 113-116, April 1986.

[3] H. A. Murthy, K. V. M. Murthy, and B. Yegnanarayana, "Formant extraction from Fourier transform phase," in *Proc. International Conf. Acoustics, Speech, Signal Processing*, pp. 484-487, May 1989.

[4] P. Smaragdis, "Discovering auditory objects through nonnegativity constraints," in *Proc. ISCA Tutorial and Research Workshop Statistical Perceptual Audio Process.*, Jeju Island, Korea, 2004.

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

[6] J. Le Roux, N. Ono, and S. Sagayama, "Explicit consistency constraints for STFT spectrograms and their application to phase reconstruction," in *Proc. SAPA*, Sep. 2008.

[7] L. D. Alsteris and K. K. Paliwal, "Short-time phase spectrum in speech processing: A review and some experimental results," *Digital Signal Process*, vol. 17, no. 5, pp. 578-616, 2007.

[8] D.L. Wang and J.S. Lim, "The unimportance of phase in speech enhancement," in *IEEE Trans. Acoust. Speech Signal Process*, vol. 30, no. 4, pp. 679-681, 1982.

[9] K. K. Paliwal and L. Alsteris, "Usefulness of phase spectrum in human speech perception," in *Proc. Eur. Conf. Speech Communication and Technology (Eurospeech)*, Geneva, Switzerland, pp. 2117-2120, Sep. 2003.

[10] G. Shi, M. M. Shanechi, and P. Aarabi, "On the importance of phase in human speech recognition,"*IEEE Trans. Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1867–1874, 2006.

[11] F. Léonard, "Phase spectrogram and frequency spectrogram as new diagnostic tools," *Mechanical Systems and Signal Processing*, vol. 21, pp. 125-137, 2007.

[12] J. E. Dennis Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[13] T. Heittola, *Automatic Classification of Music Signals*. Tampere University of Technology: Master of Science Thesis, 2003.

[14] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, and N. Dahlgren, *DARPA TIMIT acoustic-phonetic continuous speech corpus*. Gaithersburg, MD: Technical Report NISTIR 4930, National Institute of Standards and Technology, 1993.

[15] T. M. Cover and J. A. Thomas, *Elements of Information Theory, 2nd Edition*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2006.

[16] A. Papoulis, *Probability, Random Variables and Stochastic Processes, 3rd Edition*. McGraw-Hill Inc., 1991.

[17] P. A. Viola, "Alignment by Maximization of Mutual Information," *Massachusetts Institute of Technology, Artificial Intelligence Laboratory, Technical Report*, no. 1548, Jun. 1995.