A DEEP AUTO-ENCODER BASED LOW-DIMENSIONAL FEATURE EXTRACTION FROM FFT SPECTRAL ENVELOPES FOR STATISTICAL PARAMETRIC SPEECH SYNTHESIS

Shinji Takaki[1], *Junichi Yamagishi*[1,2]

[1]National Institute of Informatics, Japan.
[2]The Centre for Speech Technology Research (CSTR), University of Edinburgh, United Kingdom.

ABSTRACT

In the state-of-the-art statistical parametric speech synthesis system, a speech analysis module, e.g. STRAIGHT spectral analysis, is generally used for obtaining accurate and stable spectral envelopes, and then low-dimensional acoustic features extracted from obtained spectral envelopes are used for training acoustic models. However, a spectral envelope estimation algorithm used in such a speech analysis module includes various processing derived from human knowledge. In this paper, we present our investigation of deep auto-encoder based, non-linear, data-driven and unsupervised low-dimensional feature extraction using FFT spectral envelopes for statistical parametric speech synthesis. Experimental results showed that a text-to-speech synthesis system using deep auto-encoder based low-dimensional feature extraction from FFT spectral envelopes is indeed a promising approach.

***Index Terms*—** Statistical parametric speech synthesis, Deep auto-encoder, Spectral envelope, Vocoder

1. INTRODUCTION

Research on statistical parametric speech synthesis (SPSS) has been significantly advanced due to deep neural networks (DNNs) with many hidden layers. For instance, DNNs have been applied to acoustic modeling. Zen et al. [1] use DNN to learn the relationship between input texts and extracted features instead of decision tree-based state tying. Restricted Boltzmann machines or deep belief networks have been used to model the output probabilities of hidden Markov model (HMM) states instead of Gaussian mixture models (GMMs) [2]. Recurrent neural networks and long-short term memories have been used for prosody modeling [3] and acoustic trajectory modeling [4].

However, it is often said that averaging in SPSS still removes the spectral fine structure of natural speech. Low-dimensional spectral feature extraction from STRAIGHT spectral envelopes based on a deep auto-encoder (DAE) has been proposed for SPSS to alleviate this problem [5]. In this framework, more precise spectral features are automatically extracted in a data-driven and unsupervised way than with the standard mel-cepstrum coefficients. We have also proposed a new DNN system where spectral feature extraction, acoustic modeling and spectral post-filtering are conducted based on several DNNs [6, 7].

The DNN-based automatic speech recognition (ASR) field has several interesting challenges to extract robust features from raw inputs such as FFT spectra or raw speech waveforms recently (e.g. [8, 9]), and some papers have achieved a relative reduction in word error rates using a combination of features derived from raw waveforms and log-mel features [10]. These research results indicate that the DNN-based approaches have the potential to find more efficient acoustic features automatically than carefully designed features based on perceptual knowledge.

Motivated by the success of the above approaches, this paper focuses on a low-dimensional feature extraction from FFT spectral envelopes for SPSS. Many SPSS systems are based on advanced vocoders such as STRAIGHT [11] or WORLD [12]. One of the aims in the use of these advanced vocoders is to obtain accurate and stable spectral envelopes as well as to synthesize a high-quality speech waveform. For instance, a spectral envelope estimation algorithm implemented in the WORLD vocoder performs F0-adaptive windowing, smoothing of the power spectrum, and spectral recovery in the quefrency domain for obtaining accurate and stable spectral envelopes [13, 14]. Such accurate and stable spectral envelopes have been demonstrated to be a good choice for SPSS. However, in the meantime, it is scientifically interesting to investigate whether or not the DNN can find better features from simple spectral amplitude representation compared to cases where acoustic features are extracted from carefully processed spectral envelopes used in the advanced vocoders. Thus, we investigate a deep auto-encoder based feature extraction from FFT spectral envelopes for SPSS and compare it with SPSS systems based on several spectral envelope estimation techniques, i.e. STRAIGHT spectral analysis and WORLD spectral analysis with low-dimensional feature extractors (mel-cepstrum analysis or deep auto-encoder).

The rest of this paper is organized as follows. Section 2 shows the related work using an auto-encoder in the

This work was supported in part by EPSRC through Programme Grant EP/I031022/1 (NST) and EP/J002526/1 (CAF). Shinji Takaki was supported in part by NAVER Labs.

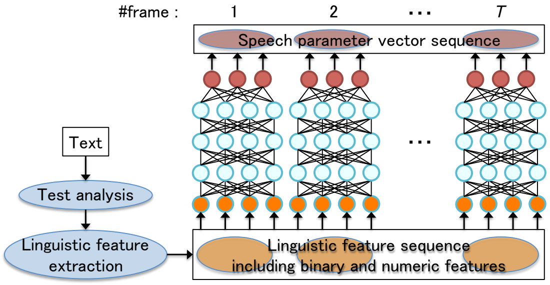

#frame :
1
2
...
T
Speech parameter vector sequence
Text
Test analysis
Linguistic feature extraction
Linguistic feature sequence including binary and numeric features
...

Fig. 1. A framework for the DNN-based acoustic model.

speech information processing. Section 3 briefly describes a DNN-based acoustic model for SPSS. In Section 4, deep auto-encoder based low-dimensional spectral parameter extraction is shown. The experimental conditions and results are shown in Section 5. Concluding remarks and future work are presented in Section 6.

2. RELATED WORK USING AN AUTO-ENCODER IN THE SPEECH INFORMATION PROCESSING

Deep auto-encoder based bottleneck features have been used by several groups for ASR [15, 16] and a deep denoising auto-encoder has also verified for noise-robust ASR [17] or reverberant ASR tasks [18, 19]. Techniques that are closely related to this paper are a spectral binary coding approach using a deep auto-encoder proposed by Deng et al. [20] and a speech enhancement approach using a deep denoising auto-encoder where Lu et al. tried to reconstruct a clean spectrum from a noisy spectrum [21]. In the field of speech synthesis, similar auto-encoder based bottleneck features were tested for excitation parameters [22, 23] and statistical parametric speech synthesizers [24, 7].

3. DNN-BASED ACOUSTIC MODEL FOR STATISTICAL PARAMETRIC SPEECH SYNTHESIS

DNN-based acoustic models representing the relationship between linguistic and speech features have been proposed for statistical parametric speech synthesis[1, 2, 3, 4]. One of the state-of-the-art DNN-based acoustic models[1] is briefly reviewed in this section.

Figure 1 illustrates a framework of the DNN-based acoustic model. In this framework, linguistic features obtained from a given text are mapped into speech parameters by a DNN. The input linguistic features are composed of binary answers to questions about linguistic contexts and numeric values such as the number of words in the current phrase, the position of the current syllable in the word, and durations of the current phoneme. In [1], the output speech parameters include spectral and excitation parameters and their time derivatives (dynamic features). By using pairs of input and output features obtained from the training dataset, the parameters of the DNN can be trained with stochastic gradient descent (SGD)[25]. Speech parameters can be predicted for an arbitrary text by a trained DNN using forward propagation.

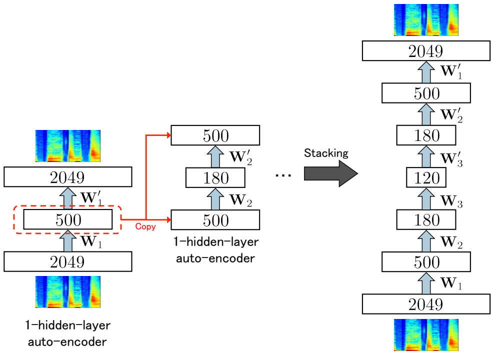


Fig. 2. Greedy layer-wise pre-training for constructing a deep auto-encoder.

4. DEEP AUTO-ENCODER BASED ACOUSTIC FEATURE EXTRACTION

4.1. Basic Auto-encoder

An auto-encoder is an artificial neural network that is used generally for learning a compressed and distributed representation of a dataset. It consists of an encoder and a decoder. The encoder in the basic one-hidden-layer auto-encoder maps an input vector $\mathbf{x}$ to a compressed hidden representation $\mathbf{y}$ as follows:

$$\mathbf{y} = f_\theta(\mathbf{x}) = s(\mathbf{W}\mathbf{x} + \mathbf{b}), \tag{1}$$

where $\theta = \{\mathbf{W}, \mathbf{b}\}$. $\mathbf{W}$ is a $m \times n$ weight matrix and $\mathbf{b}$ is a m dimension bias vector. The function s is a non-linear transformation on the linear mapping $\mathbf{W}\mathbf{x} + \mathbf{b}$. We typically use sigmoid, tanh or ReLU for the non-linear transformation. The output from the encoder is a low-dimensional representation $\mathbf{y}$, which is then passed into the decoder $g_{\theta'}$ to reconstruct back to the original dimension. The reconstruction is performed by a linear mapping followed by an arbitrary linear or non-linear function t that utilizes an $n \times m$ weight matrix $\mathbf{W}'$ and a bias vector of dimensionality n as follows:

$$\mathbf{z} = g_{\theta'}(\mathbf{y}) = t(\mathbf{W}'\mathbf{y} + \mathbf{b}'), \tag{2}$$

where $\theta' = \{\mathbf{W}', \mathbf{b}'\}$. The parameters $\{\theta, \theta'\}$ are optimized such that the reconstructed $\mathbf{z}$ is as close as possible to the original $\mathbf{x}$. The mean squared error (MSE) is typically used as the objective function for SGD to measure the distance between the input vector $\mathbf{x}$ and the reconstructed vector $\mathbf{z}$.

4.2. Deep Auto-encoder

An auto-encoder can be made deeper by stacking multiple layers of encoders and decoders to form a deep architecture. Pre-training is widely used for constructing a deep auto-encoder. In pre-training, the number of layers in a deep auto-encoder increases twice as compare to a deep neural network (DNN) when stacking each pre-trained unit. We restrict the decoding weight as the transpose of the encoding weight following [25], that is, $\mathbf{W}' = \mathbf{W}^T$ where $\mathbf{W}^T$ denotes the transpose of $\mathbf{W}$. Each layer of a deep auto-encoder can be pre-trained greedily to minimize the reconstruction loss of the data locally. Figure 2 shows the procedure for constructing a deep auto-encoder using layer-by-layer pre-training. In pre-training, a one-hidden-layer basic auto-encoder is trained and the encoding output of the locally trained layer is used as the input to the next basic auto-encoder with a smaller bottleneck layer. After all layers are pre-trained, they are stacked and fine-tuned with SGD to minimize the reconstruction error over the entire dataset. Note that mean squared error (MSE) is used as the loss function for both pre-training and fine-tuning.

A deep auto-encoder allows us to extract robust low-dimensional features automatically from high-dimensional spectral envelopes in a non-linear, data-driven and unsupervised way. In this paper, we apply the deep auto-encoder to STRAIGHT, WORLD and FFT spectral envelopes for extracting low-dimensional features, respectively[1].

5. EXPERIMENTS

We have evaluated several low-dimensional feature extraction techniques using an English database. The database that was provided for the Blizzard Challenge 2011 [26], which contains approximately 17 hours of speech data comprising 12K utterances, was used for the experiment. All data were sampled at 48 kHz. 200 sentences that are not included in the database were used as a test set.

We compared seven systems: *HMM*, *SRT-MCEP*, *SRT-DAE*, *WRD-MCEP*, *WRD-DAE*, *FFT-MCEP* and *FFT-DAE*. Table 1 shows detailed information on each system. For obtaining spectral envelopes, spectral analysis techniques implemented in STRAIGHT vocoder, WORLD vocoder or the FFT were used for each system. Here, the FFT spectral envelopes were estimated just using the F0-adaptive windowing, followed by basic FFT operation. Aperiodicity measures implemented in the STRAIGHT or WORLD vocoder were also calculated. We have used either mel-cepstrum analysis or a deep auto-encoder for extracting low-dimensional spectral features from each of the obtained spectral envelopes. For acoustic models, HMMs [27] for the system *HMM* and DNNs [1] for other systems were used. We synthesized speech samples using spectral envelopes, F0 features and aperiodicity measures using the STRAIGHT or WORLD vocoder even for FFT spectral case. With mel-cepstral analysis, predicted cepstral coefficients were converted into spectrum amplitudes to use these vocoders. With the auto-encoder, the decoder part was used to convert predicted features into spectrum amplitudes. In this paper, we didn't apply spectral enhancement techniques such as global variance [28] or post-filtering [6] to all systems to reduce factors considered in the listening test.

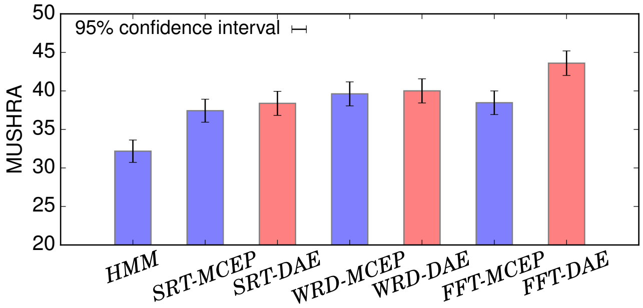


Fig. 3. Subjective results.

We trained five-hidden-layer DNN-based acoustic models for *SRT-MCEP*, *SRT-DAE*, *WRD-MCEP*, *WRD-DAE*, *FFT-MCEP* and *FFT-DAE*. The number of units in each of the hidden layers was set to 1024. Random initialization was used in a way similar to [1]. A symmetric five-hidden-layer auto-encoder was trained for *SRT-DAE*, *WRD-DAE* and *FFT-DAE*. The number of units of the hidden layers were 2049, 500, 60, 500 and 2049. We used a sigmoid function for all units of hidden and output layers of all neural networks.

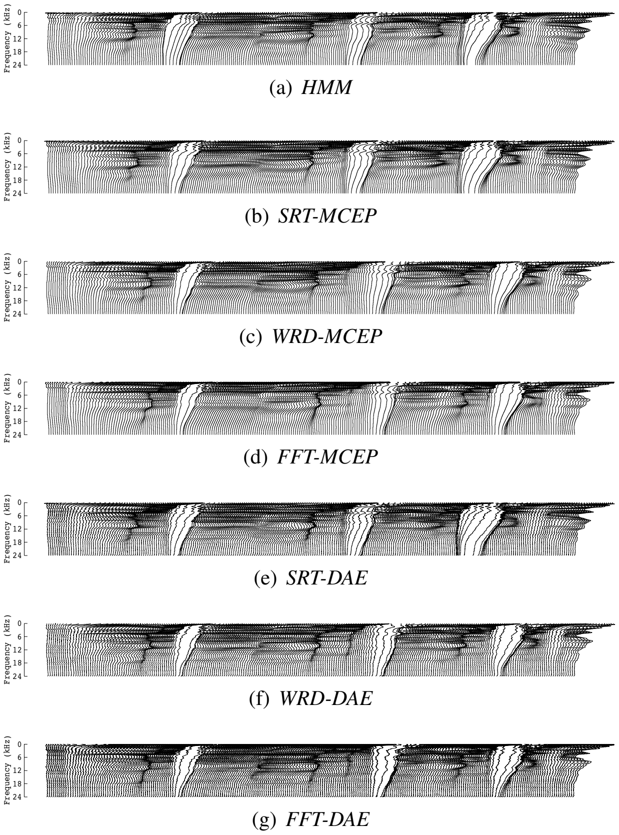
For each waveform, we extracted its frequency spectra with 2049 FFT points. For each system, 60 dimensional spectral features were extracted. Spectrum and cepstrum were both frequency-warped using the Bark scale. The feature vectors for *HMM* comprised 258 dimensions: 59 dimensional bark-cepstral coefficients (plus the 0th coefficient), $\log$ F0, 25 dimensional band aperiodicity measures, and their dynamic and acceleration coefficients. For other systems using DNN-based acoustic models, continuous $\log$ F0 interpolated linearly for unvoiced regions and voiced/unvoiced parameters were used as F0 parameters. Thus, 259 dimensional features were used as output features of the DNN. Note that $\log$ F0 was the same in all systems so that listeners could focus on differences in spectral modeling. The context-dependent labels were built using the pronunciation lexicon Combilex [29]. The linguistic features for DNN acoustic models comprised 382 dimensions. Phoneme boundaries were estimated with the HMM-based speech synthesis system. The linguistic features and spectral envelopes in the training data were pre-normalized for training DNNs. The input linguistic features were normalized to have zero-mean unit-variance, whereas the output spectral amplitudes were normalized to be within 0.0–1.0.

MUSHRA tests were conducted for subjective evaluation. Natural speech was used as a hidden top anchor reference. Twenty-four native English speaking subjects participated in the experiments. Twenty sentences were randomly selected from the test set for each subject. The experiments were carried out using professional headphones in a soundproof room.

[1] Note that the aim of this paper is to simplify spectral analysis processing used in the STRAIGHT and WORLD vocoders and use the FFT spectra as the inputs to DNN. We do not aim to replace vocoders entirely in this paper.

Table 1. Details of spectral envelope analysis, aperiodicity analysis, feature extractors, acoustic models and synthesis modules used in each system. Here, Mel-cep and DAE mean mel-cepstrum analysis and a deep auto-encoder, respectively.

Systems	*HMM*	*SRT-MCEP*	*SRT-DAE*	*WRD-MCEP*	*WRD-DAE*	*FFT-MCEP*	*FFT-DAE*
Spectral envelope	STRAIGHT	STRAIGHT	STRAIGHT	WORLD	WORLD	FFT	FFT
Aperiodicity measure	STRAIGHT	STRAIGHT	STRAIGHT	WORLD	WORLD	WORLD	WORLD
Feature Extraction	Mel-cep	Mel-cep	DAE	Mel-cep	DAE	Mel-cep	DAE
Acoustic Model	HMM	DNN	DNN	DNN	DNN	DNN	DNN
Synthesis	STRAIGHT	STRAIGHT	STRAIGHT	WORLD	WORLD	WORLD	WORLD



(a) *HMM*

(b) *SRT-MCEP*

(c) *WRD-MCEP*

(d) *FFT-MCEP*

(e) *SRT-DAE*

(f) *WRD-DAE*

(g) *FFT-DAE*

Fig. 4. Synthesized running log spectra for each technique.

5.1. Experimental results

Figure 3 shows the subjective results in the experiment. The results for natural speech were excluded from the figures to make the comparison easier.

First, we can see from the figure 3 that *HMM* was rated lower than other DNN-based speech synthesis systems. This confirms that the DNN-based acoustic models have predicted more natural acoustic features, as the previous research reported [27]. Compared with the results between the STRAIGHT vocoder (*SRT-MCEP* and *SRT-DAE*) and the WORLD vocoder (*WRD-MCEP* and *WRD-DAE*), the systems using the WORLD vocoder were rated slightly higher than STRAIGHT ones, although the differences between the two vocoders were not statistically significant. Interestingly, however, *FFT-DAE* significantly outperformed all other systems although *FFT-MCEP* was rated the lowest among systems using the WORLD vocoder.

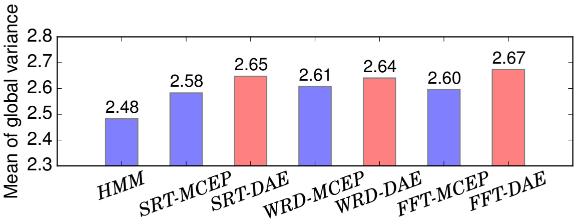


Fig. 5. Mean of global variance calculated using synthesized log spectra for all utterances in each technique.

For further analysis of the results, We plotted parts of synthesized running log spectra in each technique in figure 4. We can clearly see from the figure 4 that the systems using a deep auto-encoder, especially *FFT-DAE*, output more dynamic trajectory compared with the systems using mel-cepstrum analysis. Figure 5 shows the mean of the global variance calculated using synthesized log spectra for all test utterances in each technique. It is well known that a synthesized trajectory is often excessively smoothed due to the statistical processing, and the global variance of the synthesized trajectory tends to be smaller than that of natural speech [28]. We found that the systems using the auto-encoder have larger global variance, and this trend matches the results of our listening test results well. From figures 4 and 5, we conclude that *FFT-DAE* predicted most dynamic spectral envelopes and generated higher quality sounds. These results indicate that a deep auto-encoder extracts more efficient and effective low-dimensional acoustic features for SPSS even from raw FFT spectral envelopes in a data-driven, unsupervised way.

6. CONCLUSIONS

This paper presented our investigation on deep auto-encoder based feature extraction from raw FFT spectral envelopes for SPSS. In the experiments, we constructed seven text-to-speech synthesizers using different acoustic models, spectral analysis methods and low-dimensional feature extractors. Interestingly, a text-to-speech synthesizer using an FFT-spectral based deep auto-encoder outperformed conventional systems used in the experiment.

Applying enhancement techniques, e.g. global variance or post-filtering, to constructed speech synthesizers and the use of raw speech waveforms are our future work.

7. REFERENCES

[1] H. Zen, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," *Proceedings of ICASSP*, pp. 7962–7966, 2013.

[2] Z.-H. Ling, L. Deng, and D. Yu, "Modeling spectral envelopes using restricted Boltzmann machines and deep belief networks for statistical parametric speech synthesis," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, pp. 2129–2139, 2013.

[3] Y. Fan, Y. Qian, F. Xie, and F. K. Soong, "TTS synthesis with bidirectional LSTM based recurrent neural networks," *Proceedings of Interspeech*, pp. 1964–1968, 2014.

[4] R. Fernandez, A. Rendel, B. Ramabhadran, and R. Hoory, "Prosody contour prediction with long short-term memory, bi-directional, deep recurrent neural networks," *Proceedings of Interspeech*, pp. 2268–2272, 2014.

[5] S. Takaki, W. Zhenzhou, and J Yamagishi, "A function-wise pre-training technique for constructing a deep neural network based spectral model in statistical parametric speech synthesis," *Machine Learning in Spoken Language Processing (MLSLP)*, 2015.

[6] L.-H. Chen, T. Raitio, C. Valentini-Botinhao, J. Yamagishi, and Z.-H. Ling, "DNN-based stochastic postfilter for HMM-based speech synthesis," *Proceedings of Interspeech*, pp. 1954–1958, 2014.

[7] S. Takaki, S.-J. Kim, J. Yamagishi, and j.-J Kim, "Multiple feed-forward deep neural networks for statistical parametric speech synthesis," *Proceedings of Interspeech*, pp. 2242–2246, 2015.

[8] Z. T*ü*ske, P. Golik, R. Schl*ü*ter, and H. Ney, "Acoustic modeling with deep neural networks using raw time signal for lvcsr," *Proceedings of Interspeech*, pp. 890–894, 2014.

[9] D. Palaz, M. Magimai.-Doss, and Collobert R., "Convolutional neural networks-based continuous speech recognition using raw speech signal2, journal =," .

[10] T. N. Sainath, R. J. Weiss, A. Senior, K. W. Wilson, and O. Vinyals, "Learning the speech front-end with raw waveform cldnns," *Proceedings of Interspeech*, pp. 1–5, 2015.

[11] H. Kawahara, I. Masuda-Katsuse, and A. Cheveigne, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, pp. 187–207, 1999.

[12] M. Morise, "An attempt to develop a singing synthesizer by collaborative creation," *the Stockholm Music Acoustics Conference 2013 (SMAC2013)*, pp. 289–292, 2015.

[13] M. Morise, "Cheaptrick, a spectral envelope estimator for high-quality speech synthesis,," *Speech Communication*, vol. 67, pp. 1–7, 2015.

[14] M. Morise, "Error evaluation of an f0-adaptive spectral envelope estimator in robustness against the additive noise and f0 error," *IEICE transactions on information and systems*, vol. E98-D, no. 7, pp. 1405–1408, 2015.

[15] T. N. Sainath, B. Kingsbury, and B. Ramabhadran, "Auto-encoder bottleneck features using deep belief networks," *Proceedings of ICASSP*, pp. 4153–4156, 2012.

[16] J. Gehring, Y. Miao, F. Metze, and A. Waibel, "Extracting deep bottleneck features using stacked auto-encoders," *Proceedings of ICASSP*, pp. 3377–3381, 2013.

[17] A. L. Maas, Q. V. Le, T. M. ONeil, O. Vinyals, P. Nguyen, and A. Ng Andrew, "Recurrent neural networks for noise reduction in robust ASR," *Proceedings of Interspeech*, pp. 22–25, 2012.

[18] T. Ishii, H. Komiyama, T. Shinozaki, Y. Horiuchi, and S Kuroiwa, "Reverberant speech recognition based on denoising autoencoder," *Proceedings of Interspeech*, pp. 3512–3516, 2013.

[19] X. Feng, Y. Zhang, and J. Glass, "Speech feature denoising and dereverberation via deep autoencoders for noisy reverberant speech recognition," *Proceedings of ICASSP*, pp. 1778–1782, 2014.

[20] L. Deng, M. Seltzer1, D. Yu, A. Acero, A. Mohamed, and G. Hinton, "Binary coding of speech spectrograms using a deep auto-encoder," *Proceedings of Interspeech*, pp. 1692–1695, 2010.

[21] X. Lu, Y. Tsao, S. Matsuda1, and C. Hori, "Speech enhancement based on deep denoising autoencoder," *Proceedings of Interspeech*, pp. 436–440, 2013.

[22] R. Vishnubhotla, S. Fernandez and B. Ramabhadran, "An autoencoder neural-network based low-dimensionality approach to excitation modeling for hmm-based text-to-speech," *Proceedings of ICASSP*, pp. 4614–4617, 2010.

[23] T. Raitio, A. Suni, L. Juvela, M. Vainio, and P. Alku, "Deep neural network based trainable voice source model for synthesis of speech with varying vocal effort," *Proceedings of Interspeech*, pp. 1969–1973, 2014.

[24] P. K. Muthukumar and Black. A., "A deep learning approach to data-driven parameterizations for statistical parametric speech synthesis," *CoRR*, vol. abs/1409.8558, 2014.

[25] G. E. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science 28*, vol. 313, no. 5786, pp. 504–507, 2006.

[26] S King and V. Karaiskos, "The blizzard challenge 2011," 2011.

[27] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, pp. 1039–1064, 2009.

[28] T. Toda and K. Tokuda, "Speech parameter generation algorithm considering global variance for HMM-based speech synthesis," *Proceedings of Interspeech 2005*, pp. 2801–2804, 2005.

[29] K. Richmond, R. Clark, and S. Fitt, "On generating combilex pronunciations via morphological analysis," *Proceedings of Interspeech*, pp. 1974–1977, 2010.