TRAINING ALGORITHM TO DECEIVE ANTI-SPOOFING VERIFICATION FOR DNN-BASED SPEECH SYNTHESIS

Yuki Saito, Shinnosuke Takamichi, and Hiroshi Saruwatari

Graduate School of Information Science and Technology, The University of Tokyo,
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
Email: {yuuki_saito, shinnosuke_takamichi, hiroshi_saruwatari}@ipc.i.u-tokyo.ac.jp

ABSTRACT

This paper proposes a novel training algorithm for high-quality Deep Neural Network (DNN)-based speech synthesis. The parameters of synthetic speech tend to be over-smoothed, and this causes significant quality degradation in synthetic speech. The proposed algorithm takes into account an Anti-Spoofing Verification (ASV) as an additional constraint in the acoustic model training. The ASV is a discriminator trained to distinguish natural and synthetic speech. Since acoustic models for speech synthesis are trained so that the ASV recognizes the synthetic speech parameters as natural speech, the synthetic speech parameters are distributed in the same manner as natural speech parameters. Additionally, we find that the algorithm compensates not only the parameter distributions, but also the global variance and the correlations of synthetic speech parameters. The experimental results demonstrate that 1) the algorithm outperforms the conventional training algorithm in terms of speech quality, and 2) it is robust against the hyper-parameter settings.

Index Terms— DNN-based speech synthesis, anti-spoofing verification, training algorithm, generative adversarial training, multi-task learning

1. INTRODUCTION

Statistical parametric speech synthesis [1] is a technique that aims to generate natural-sounding synthetic speech. Hidden Markov Models (HMMs) [2] and Deep Neural Networks (DNNs) [3] are used as the acoustic models for speech synthesis, and these acoustic models are trained with several training algorithms such as the maximum likelihood criterion [2] and Minimum Generation Error (MGE) criterion [4, 5]. Acoustic modeling techniques for generating high-quality speech are widely studied since they can be used for text-to-speech, voice conversion, and multimodal speech synthesis. However, the speech parameters generated from these models tend to be over-smoothed, and the quality of their speech is still low compared with that of natural speech [1, 6].

One promising way to improve speech quality is to compensate for the divergence between the natural and generated speech parameters. The parameter distributions of natural and synthetic speech are significantly different [7], but quality improvements can be had by transforming the synthetic speech parameters so that their distribution is close to that of natural speech parameters. This can be done by, for example, modeling the probability distributions in a parametric [8] or non-parametric [9] way in the training stage, and then, generating or transforming the synthetic speech parameters by using the distributions. The more effective approach is to use analytically derived features correlated to the quality degradation of the synthetic speech. Global Variance (GV) [8] and Modulation Spectrum (MS) [10] are well-known examples. They work as an additional term in the training/synthesis stage [11, 12]. Nose and Ito [13] and Takamichi et al. [11] proposed methods that reduce the difference between Gaussian distributions of natural and generated GV and MS. However, quality degradation is still a critical problem.

In order to address this quality problem, this paper proposes a novel training algorithm that uses an Anti-Spoofing Verification (ASV) for statistical parametric speech synthesis. The ASV is a discriminator trained to distinguish natural and synthetic speech. Since the acoustic models for speech synthesis are trained to deceive the ASV, the parameter distribution of the synthetic speech is close to that of natural speech. The training criterion is the weighted sum of the conventional training criterion and the spoofing rate for the ASV, and the training is simply done using a backpropagation algorithm by using DNNs for speech synthesis and ASV. Furthermore, the training algorithm can be regarded as a generalization of the conventional approaches; i.e., 1) we use not only analytically derived features (e.g., GV and MS), but also automatically derived (DNN-derived) features as the compensated features, and 2) we model a probability distribution that is more complicated than the conventional Gaussian distribution. We conducted an experimental evaluation to demonstrate the effectiveness of our algorithm. The experimental results demonstrate that 1) the algorithm outperforms the conventional MGE training in terms of speech quality and 2) it works comparably robustly against the hyper-parameter settings.

2. CONVENTIONAL TRAINING ALGORITHM

2.1. DNNs as acoustic models

In DNN-based speech synthesis [14], acoustic models that represent a relationship between linguistic features and speech parameters consist of layered hierarchical networks. To construct the models, we minimize a loss function calculated using natural and synthetic speech parameters. Let $\boldsymbol{c}$ be a natural speech parameter sequence $\left[\boldsymbol{c}_1^\top, \cdots, \boldsymbol{c}_t^\top, \cdots, \boldsymbol{c}_T^\top\right]^\top$, and $\hat{\boldsymbol{c}}$ be a synthetic speech parameter sequence $\left[\hat{\boldsymbol{c}}_1^\top, \cdots, \hat{\boldsymbol{c}}_t^\top, \cdots, \hat{\boldsymbol{c}}_T^\top\right]^\top$, where t and T denote the frame index and total frame length, respectively. $\boldsymbol{c}_t = \left[c_t(1), \cdots, c_t(D)\right]^\top$ is a D-dimensional speech parameter vector at frame t. As in [14], the acoustic models described here predict static and dynamic speech feature vectors, $\hat{\boldsymbol{o}}_t = \left[\hat{\boldsymbol{c}}_t^\top, \Delta\hat{\boldsymbol{c}}_t^\top, \Delta\Delta\hat{\boldsymbol{c}}_t^\top\right]^\top$, at frame t.

2.2. Minimum Generation Error (MGE) training [5]

The acoustic models are trained using the Minimum Generation Error (MGE) training algorithm [5]. The loss function $L_{\mathrm{G}}(\boldsymbol{c}, \hat{\boldsymbol{c}})$ is defined as the mean squared error between natural and synthetic speech

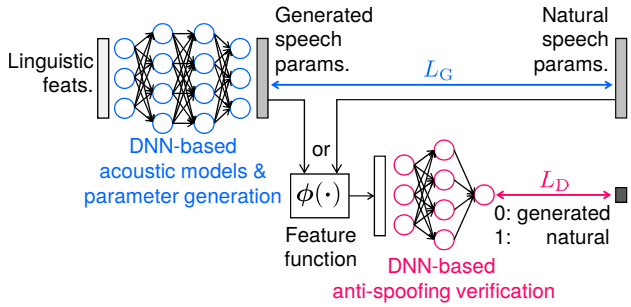

Fig. 1. Calculation of loss function in proposed algorithm.

parameters as follows:

$$L_{\mathrm{G}}\left(\boldsymbol{c}, \hat{\boldsymbol{c}}\right) = \frac{1}{T}\left(\hat{\boldsymbol{c}} - \boldsymbol{c}\right)^{\top}\left(\hat{\boldsymbol{c}} - \boldsymbol{c}\right) = \frac{1}{T}\left(\boldsymbol{R}\hat{\boldsymbol{o}} - \boldsymbol{c}\right)^{\top}\left(\boldsymbol{R}\hat{\boldsymbol{o}} - \boldsymbol{c}\right), \quad (1)$$

where $\hat{\boldsymbol{o}} = [\hat{\boldsymbol{o}}_1^{\top}, \cdots, \hat{\boldsymbol{o}}_t^{\top}, \cdots, \hat{\boldsymbol{o}}_T^{\top}]^{\top}$ is the static and dynamic speech feature sequence predicted by the acoustic models. $\boldsymbol{R}$ is a DT-by-$3DT$ matrix given as

$$\boldsymbol{R} = \left(\boldsymbol{W}^{\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{W}\right)^{-1}\boldsymbol{W}^{\top}\boldsymbol{\Sigma}^{-1}, \quad (2)$$

where $\boldsymbol{W}$ is a $3DT$-by-DT matrix for calculating dynamic features [2] and $\boldsymbol{\Sigma} = \mathrm{diag}\left[\boldsymbol{\Sigma}_1, \cdots, \boldsymbol{\Sigma}_t, \cdots, \boldsymbol{\Sigma}_T\right]$ is a $3DT$-by-$3DT$ covariance matrix, where $\boldsymbol{\Sigma}_t$ is a $3D$-by-$3D$ covariance matrix at frame t. $\boldsymbol{\Sigma}$ is separately estimated using training data.

The model parameters (e.g., weight or bias of DNNs) are estimated using the backpropagation algorithm using the gradients of $L\left(\boldsymbol{c}, \hat{\boldsymbol{c}}\right)$ by $\boldsymbol{o}$.

3. PROPOSED TRAINING ALGORITHM

3.1. Anti-Spoofing Verification (ASV) [15]

In order to distinguish natural speech and synthetic speech, an ASV discriminator is trained using the speech parameters (or speech waveforms). In DNN-based ASV (e.g., [16]), after applying the feature function $\boldsymbol{\phi}\left(\cdot\right)$ to the input speech parameters, the DNN outputs the posterior probability $D\left(\boldsymbol{\phi}\left(\cdot\right)\right)$ that the input is natural speech. Features that distinguish natural and synthetic speech parameters are often calculated at this point. Here, frame-wise speech parameters are directly used for the ASV, i.e., $\boldsymbol{\phi}\left(\boldsymbol{c}_t\right) = \boldsymbol{c}_t$. The loss function for the ASV is defined as the cross-entropy function:

$$L_{\mathrm{D}}\left(\boldsymbol{c}, \hat{\boldsymbol{c}}\right) = L_{\mathrm{D},1}\left(\boldsymbol{c}\right) + L_{\mathrm{D},0}\left(\hat{\boldsymbol{c}}\right), \quad (3)$$

$$L_{\mathrm{D},1}\left(\boldsymbol{c}\right) = -\frac{1}{T}\sum_{t=1}^{T}\log D\left(\boldsymbol{c}_t\right), \quad (4)$$

$$L_{\mathrm{D},0}\left(\hat{\boldsymbol{c}}\right) = -\frac{1}{T}\sum_{t=1}^{T}\log\left(1 - D\left(\hat{\boldsymbol{c}}_t\right)\right). \quad (5)$$

where $L_{\mathrm{D},1}\left(\boldsymbol{c}\right)$ and $L_{\mathrm{D},0}\left(\hat{\boldsymbol{c}}\right)$ are the loss functions for natural and synthetic speech parameters, respectively. The backpropagation algorithm is used to train the DNN to output 1 for natural speech and 0 for synthetic speech.

3.2. Training algorithm to deceive ASV

Here, we describe a novel training algorithm to deceive the ASV. Fig. 1 illustrates the computation procedure of the loss function.

The loss function for speech synthesis is

$$L\left(\boldsymbol{c}, \hat{\boldsymbol{c}}\right) = L_{\mathrm{G}}\left(\boldsymbol{c}, \hat{\boldsymbol{c}}\right) + \omega_{\mathrm{D}}\frac{E_{L_{\mathrm{G}}}}{E_{L_{\mathrm{D}}}}L_{\mathrm{D},1}\left(\hat{\boldsymbol{c}}\right), \quad (6)$$

where $E_{L_{\mathrm{G}}}$ and $E_{L_{\mathrm{D}}}$ denote the expectation values of $L_{\mathrm{G}}\left(\boldsymbol{c}, \hat{\boldsymbol{c}}\right)$ and $L_{\mathrm{D},1}\left(\hat{\boldsymbol{c}}\right)$, respectively. Their ratio, $E_{L_{\mathrm{G}}}/E_{L_{\mathrm{D}}}$ is the scale normalization term between $L_{\mathrm{G}}\left(\boldsymbol{c}, \hat{\boldsymbol{c}}\right)$ and $L_{\mathrm{D},1}\left(\hat{\boldsymbol{c}}\right)$, and the hyperparameter ω_{D} controls the weight of the second term. When $\omega_{\mathrm{D}} = 0$, the loss function is equivalent to the conventional MGE training, and when $\omega_{\mathrm{D}} = 1$, $L_{\mathrm{G}}\left(\boldsymbol{c}, \hat{\boldsymbol{c}}\right)$ and $L_{\mathrm{D},1}\left(\hat{\boldsymbol{c}}\right)$ have equal weights. $L_{\mathrm{D},1}\left(\hat{\boldsymbol{c}}\right)$ lets the synthetic speech parameters be recognized as natural speech in the ASV step, and it minimizes the Jensen-Shannon divergence between the distributions of the natural and synthetic speech parameters [17]. Therefore, this loss function not only minimizes the generation error but also makes the distribution of the synthetic speech parameters close to that of natural speech parameters.

After training the acoustic models to deceive the ASV, the ASV is retrained to distinguish natural and synthetic speech. The final acoustic models are estimated by iterating these algorithms.

3.3. Relation to other methods and behavior analysis

We can choose not only analytically derived features, e.g., GV and MS, but also automatically derived features (e.g., [18]) for the ASV in our training algorithm. When an analytically derived feature is used, it must be differentiable by the input speech parameters for the backpropagation algorithm to work. Also, since only the backpropagation algorithm is used for training, a variety of DNN architectures (e.g., Long Short-Term Memory (LSTM) [19]) can be used as acoustic models and the ASV.

The proposed loss function (Eq. (6)) is a combination of a multitask learning algorithm using discriminators [20] and a generative adversarial training [17]. In defining $L\left(\boldsymbol{c}, \hat{\boldsymbol{c}}\right) = L_{\mathrm{D},1}\left(\hat{\boldsymbol{c}}\right)$, the loss function is equivalent to that for generative adversarial training [17]. In other words, our algorithm can be regarded as generative adversarial training with referred input and output parameters [21].

As described above, our algorithm makes the distribution of the synthetic speech parameters close to that of natural speech parameters. Since we perform generative adversarial training with DNNs, our algorithm comes to have a more complicated probability distribution than the conventional Gaussian distribution. Fig. 2 plots natural and synthetic speech parameters with several mel-cepstral coefficient pairs. Whereas the parameters of the conventional algorithm are narrowly distributed, those of the proposed algorithm are widely distributed, the same as the natural speech parameters. Moreover, we can see that the proposed algorithm more affects the distribution of the higher order of the mel-cepstral coefficients.

Here, one can explore what components (e.g., analytically derived features and intuitive reasons [22]) the algorithm changes. Fig. 3 plots the averaged GVs of natural and synthetic speech parameters. We can see that the GV created by the proposed algorithm is closer to the natural GV than is the one produced by the conventional algorithm. Then, we calculated a Maximal Information Coefficient (MIC) [23] to quantify a correlation between each dimension of the mel-cepstral coefficients. Fig. 4 shows the results. As reported in [7], we can see that there are weak correlations between dimensions of the natural speech parameters, whereas strong correlations are observed between those of generated speech parameters of the MGE training. Meanwhile, the generated mel-cepstral coefficients of our algorithm have weak correlations than those of the MGE training. These results suggest that the proposed algorithm

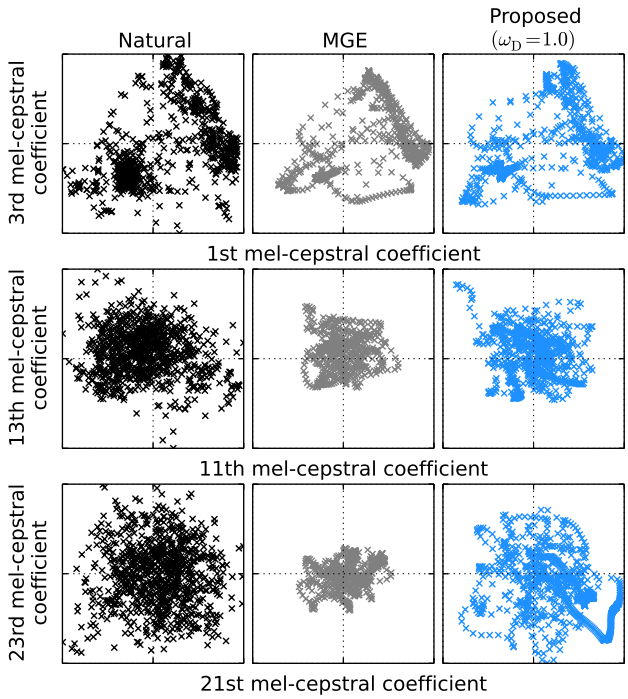

Fig. 2. Scatter plots of mel-cepstral coefficients with several pairs of dimensions. From the left, the figures correspond to natural speech, the conventional MGE algorithm, and the proposed algorithm ($\omega_{\mathrm{D}} = 1.0$), respectively. These mel-cepstral coefficients were extracted from one utterance of the evaluation data.

compensates not only the GV of the synthetic speech parameter, but also the correlation between each dimension of the parameters.

4. EXPERIMENTAL EVALUATION

4.1. Experimental conditions

We used speech data of a male speaker taken from the ATR Japanese speech database [24]. The speaker uttered 503 phonetically balanced sentences. We used 450 sentences (subsets A to I) for the training and 53 sentences (subset J) for the evaluation. Speech signals were sampled at a rate of 16 kHz, and the shift length was set to 5 ms. The 0th-through-24th mel-cepstral coefficients were used as a spectral parameter and F_0 and 5 band-aperiodicity [25, 26] were used as excitation parameters. The STRAIGHT analysis-synthesis system [27] was used for the parameter extraction and the waveform synthesis. To improve training accuracy, speech parameter trajectory smoothing [28] with a 50 Hz cutoff modulation frequency was applied to the spectral parameters in the training data. Linguistic features of contexts consisted of 274-dimensional vector including phonemes, mora position, accent type, frame position in a phoneme, and so on. We only used some of the prosody-related features because we predicted only the spectral parameters in this experiment. In the training phase, spectral features were normalized to have zero-mean unit-variance, and 80% of the silence frames were removed from the training data in order to reduce the computational cost.

The DNN architectures for acoustic models and ASV were Feed-Forward networks. The architecture for the acoustic models included 3 $\times$ 400-unit Rectified Linear Unit (ReLU) [29] hidden

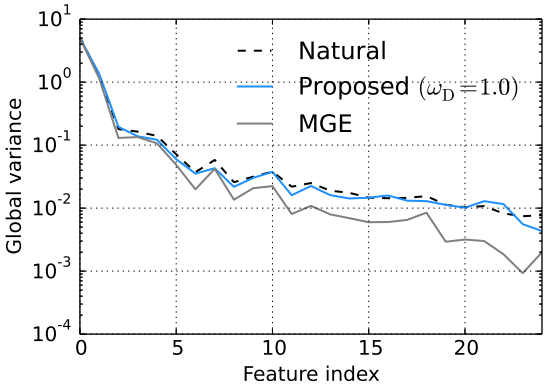

Fig. 3. Averaged GVs of synthetic mel-cepstral coefficients.

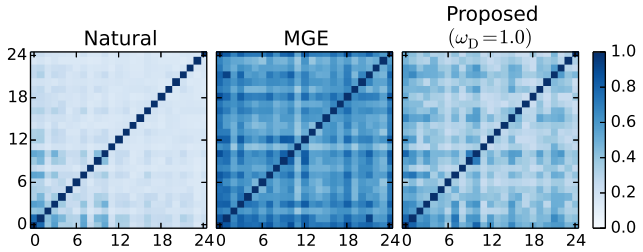

Fig. 4. MICs of natural and synthetic mel-cepstral coefficients. The MIC ranges from 0.0 to 1.0, and the two valuables with a strong correlation have a value closer to 1.0. From the left, the figures correspond to natural speech, the conventional MGE algorithm, and the proposed algorithm ($\omega_{\mathrm{D}} = 1.0$), respectively. These MICs were calculated from one utterance of the evaluation data.

layers and a 75-unit linear output layer. The architecture for the ASV included 2 $\times$ 200-unit ReLU hidden layers and a one-unit sigmoid output layer. The acoustic models output static and dynamic mel-cepstral coefficients (75-dim.) frame by frame. The ASV input static mel-cepstral coefficients (25-dim.) frame by frame. We used AdaGrad [30] as the optimization algorithm, setting the learning rate to 0.01. F_0, band-aperiodicity, and duration of natural speech were used for the speech waveform synthesis.

In the training phase, we performed the training algorithm [14] frame-by-frame for the initialization of acoustic models; then we performed the conventional MGE training [5] with 25 iterations. Here, "iteration" means using all the training data (450 utterances) once for training.

The ASV was initialized using natural speech and synthetic speech after the MGE training. The number of iterations for the ASV initialization was 5. The proposed training and ASV re-training were performed with 25 iterations. At each iteration step, the proposed training algorithm updated the acoustic model parameters by using all the training data. The ASV re-training was performed using speech parameters generated with the updated acoustic models. Expectation values $E_{L_{\mathrm{G}}}$ and $E_{L_{\mathrm{D}}}$ were estimated at each iteration step.

In order to evaluate our algorithm, we calculated the parameter generation loss defined in Eq. (1) and spoofing rate of the synthetic speech. The spoofing rate is the number of spoofing synthetic speech parameters divided by the total number of synthetic speech parameters in the evaluation data. Here, "spoofing synthetic speech parameter" indicates a parameter for which the ASV recognized the synthetic speech as natural speech. The ASV for calculating the

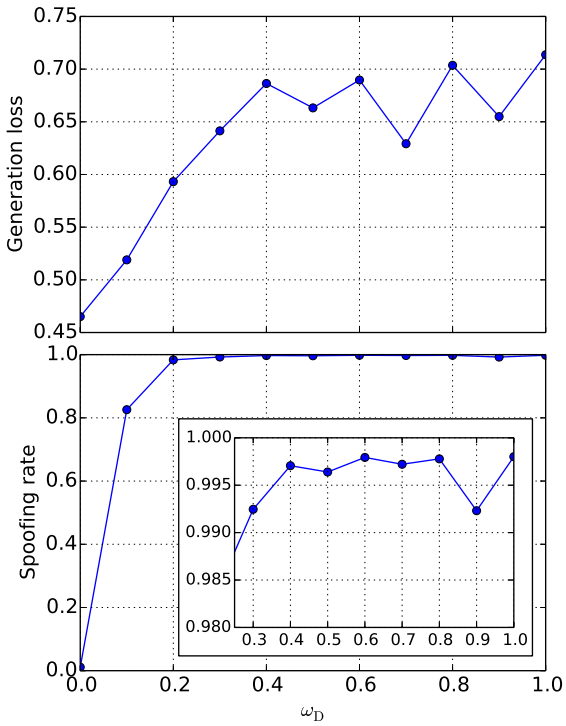

Fig. 5. Parameter generation loss (above) and spoofing rate (below) for various ω_D.

spoofing rates was constructed using natural speech parameters and synthetic speech parameters of the conventional MGE training. The generation loss and spoofing rates were first calculated with various hyper-parameter ω_D settings. Then, we conducted a subjective evaluation of the quality of the synthetic speech.

4.2. Objective evaluation

Fig. 5 shows the results for the generation loss and spoofing rate. As ω_D increases from 0.0, the generation loss monotonically increases, but from 0.4, we cannot see any tendency. On the other hand, the spoofing rate significantly increases as ω_D increases from 0.0 to 0.2; from 0.2, the value does not so vary. These results demonstrate that the proposed training algorithm makes the generation loss worse but can train the acoustic models to deceive the ASV.

4.3. Subjective evaluation

A preference test (AB test) was conducted to evaluate the quality of speech produced by the algorithm. We generated speech samples with three methods:

MGE: conventional MGE (= proposed ($\omega_D = 0.0$))

Proposed ($\omega_D = 0.3$)**:** spoofing rate > 0.99

Proposed ($\omega_D = 1.0$)**:** standard setting of ω_D

We presented every pair of synthetic speech of the three sets in random order and forced listeners to select the speech sample that sounded better in quality. Eight listeners participated in the assessment.

Fig. 6 shows the results. The resulting scores of "Proposed" were much higher than those of "MGE." Thus, our algorithm yielded a remarkable improvement in terms of speech quality. Moreover, we can see from the results that there is no significant difference

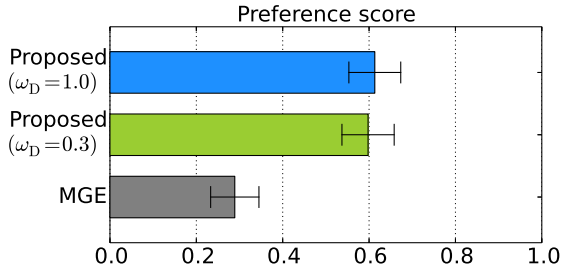

Fig. 6. Preference scores of speech quality with 95% confidence intervals.

between "Proposed ($\omega_D = 0.3$)" and "Proposed ($\omega_D = 1.0$)." Since the objective results shown in Fig. 5 do not vary much when ω_D is between 0.3 and 1.0, it is expected that the speech quality will be almost the same between 0.3 and 1.0. Therefore, this result suggests that our algorithm works comparably robustly against the hyper-parameter setting.

5. CONCLUSION

This paper proposed a novel training algorithm for high-quality Deep Neural Network (DNN)-based speech synthesis. An Anti-Spoofing Verification (ASV), which distinguishes natural speech and synthetic speech, is used to train the acoustic models of speech synthesis. The acoustic models are trained to not only minimize the generation loss but also make the parameter distribution of the synthetic speech parameters close to that of natural speech parameters. We found that our algorithm compensated not only global variance but also correlation of synthetic speech parameters. The experimental results showed significant improvements in terms of speech quality. Moreover, it was demonstrated that the algorithm works comparably robustly against the hyper-parameter settings. In the future, we will use the features that are effective for the ASV, further investigate the behavior in relation to the hyper-parameter settings, and devise ASV models with temporal [31] and linguistic [21] dependencies.

Acknowledgements: Part of this work was supported by ImPACT Program of Council for Science, Technology and Innovation (Cabinet Office, Government of Japan), SECOM Science and Technology Foundation, and JSPS KAKENHI Grant Number 16H06681.

6. REFERENCES

[1] H. Zen, K. Tokuda, and A. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039–1064, 2009.

[2] K. Tokuda, Y Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on hidden Markov models," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1234–1252, 2013.

[3] Z. H. Ling, S. Y. Kang, H. Zen, A. Senior, M. Schuster, X. J. Qian, H. Meng, and L. Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 35–52, 2015.

[4] Y. J. Wu and R. H. Wang, "Minimum generation error training for HMM-based speech synthesis," in *Proc. ICASSP*, Toulouse, France, May 2006, pp. 89–92.

[5] Z. Wu and S. King, "Improving trajectory modeling for DNN-based speech synthesis by using stacked bottleneck features

and minimum trajectory error training," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 7, pp. 1255–1265, 2016.

[6] T. Toda, L. H. Chen, D. Saito, F. Villavicencio, M. Wester, Z. Wu, and J. Yamagishi, "The Voice Conversion Challenge 2016," in *Proc. INTERSPEECH*, California, U.S.A., Sep. 2016, pp. 1632–1636.

[7] Y. Ijima, T. Asami, and H. Mizuno, "Objective evaluation using association between dimensions within spectral features for statistical parametric speech synthesis," in *Proc. INTERSPEECH*, California, U.S.A., Sep. 2016, pp. 337–341.

[8] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[9] Y. Ohtani, M. Tamura, M. Morita, T. Kagoshima, and M. Akamine, "Histogram-based spectral equalization for HMM-based speech synthesis using mel-LSP," in *Proc. INTERSPEECH*, Portland, U.S.A., Sep. 2012.

[10] S. Takamichi, T. Toda, A. W. Black, G. Neubig, S. Sakti, and S. Nakamura, "Postfilters to modify the modulation spectrum for statistical parametric speech synthesis," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 755–767, 2016.

[11] S. Takamichi, T. Toda, A. W. Black, and S. Nakamura, "Modulation spectrum-constrained trajectory training algorithm for GMM-based voice conversion," in *Proc. ICASSP*, Brisbane, Australia, Apr. 2015, pp. 4859–4863.

[12] K. Hashimoto, K. Oura, Y. Nankaku, and K.Tokuda, "Trajectory training considering global variance for speech synthesis based on neural networks," in *Proc. ICASSP*, Shanghai, China, Mar. 2016, pp. 5600–5604.

[13] T. Nose and A. Ito, "Analysis of spectral enhancement using global variance in HMM-based speech synthesis," in *Proc. INTERSPEECH*, MAX Atria, Singapore, May 2014, pp. 2917–2921.

[14] H. Zen, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Proc. ICASSP*, Vancouver, Canada, May 2013, pp. 7962–7966.

[15] Z. Wu, P. L. D. Leon, C. Demiroglu, A. Khodabakhsh, S. King, Z. Ling, D. Saito, B. Stewart, T. Toda, M. Wester, and J. Yamagishi, "Anti-spoofing for text-independent speaker verification: An initial database, comparison of countermeasures, and human performance," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 768–783, 2016.

[16] N. Chen, Y. Qian, H. Dinkel, B. Chen, and K. Yu, "Robust deep feature for spoofing detection — the SJTU system for ASVspoof 2015 Challenge," in *Proc. INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 2097–2101.

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27*, pp. 2672–2680. 2014.

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

[19] H. Zen and H. Sak, "Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis," in *Proc. ICASSP*, Brisbane, Australia, Apr. 2015, pp. 4470–4474.

[20] B. Huang, D. Ke, H. Zheng, B. Xu, Y. Xu, and K. Su, "Multi-task learning deep neural networks for speech feature denoising," in *Proc. INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 2464–2468.

[21] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text-to-image synthesis," in *Proc. ICML*, 2016, pp. 1060–1069.

[22] T. R. Marco, S. Sameer, and G. Carlos, ""Why should I trust you?": Explaining the predictions of any classifier," in *Proc. KDD*, San Francisco, U.S.A., Aug. 2016, pp. 1135–1164.

[23] D. N. Reshef, Y. A. Reshef, H. K. Finucane, S. R. Grossman, G. McVean, P. J. Turnbaugh, E. S. Lander, M. Mitzenmacher, and P. C. Sabeti, "Detecting novel associations in large data sets," vol. 334, no. 6062, pp. 1518–1524, 2011.

[24] M. Abe, Y. Sagisaka, T. Umeda, and H. Kuwabara, "ATR technical repoart," , no. TR-I-0166M, 1990.

[25] H. Kawahara, Jo Estill, and O. Fujimura, "Aperiodicity extraction and control using mixed mode excitation and group delay manipulation for a high quality speech analysis, modification and synthesis system STRAIGHT," in *MAVEBA 2001*, Firentze, Italy, Sep. 2001, pp. 1–6.

[26] Y. Ohtani, T. Toda, H. Saruwatari, and K. Shikano, "Maximum likelihood voice conversion based on GMM with STRAIGHT mixed excitation," in *Proc. INTERSPEECH*, Pittsburgh, U.S.A., Sep. 2006, pp. 2266–2269.

[27] H. Kawahara, I. Masuda-Katsuse, and A. D. 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, no. 3–4, pp. 187–207, 1999.

[28] S. Takamichi, K. Kobayashi, K. Tanaka, T. Toda, and S. Nakamura, "The NAIST text-to-speech system for the Blizzard Challenge 2015," in *Proc. Blizzard Challenge workshop*, Berlin, Germany, Sep. 2015.

[29] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. AISTATS*, Lauderdale, U.S.A., Apr. 2011, pp. 315–323.

[30] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, 2011.

[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.