Adapting and Controlling DNN-Based Speech Synthesis Using Input Codes

ADAPTING AND CONTROLLING DNN-BASED SPEECH SYNTHESIS USING INPUT CODES

Hieu-Thi Luong[1], Shinji Takaki[2], Gustav Eje Henter[2], Junichi Yamagishi[2,3]†*

[1]VNUHCM – University of Science, Ho Chi Minh City, Vietnam
[2]National Institute of Informatics, Tokyo, Japan [3]The University of Edinburgh, Edinburgh, UK

ABSTRACT

Methods for adapting and controlling the characteristics of output speech are important topics in speech synthesis. In this work, we investigated the performance of DNN-based text-to-speech systems that in parallel to conventional text input also take speaker, gender, and age codes as inputs, in order to 1) perform multi-speaker synthesis, 2) perform speaker adaptation using small amounts of target-speaker adaptation data, and 3) modify synthetic speech characteristics based on the input codes. Using a large-scale, studio-quality speech corpus with 135 speakers of both genders and ages between tens and eighties, we performed three experiments: 1) First, we used a subset of speakers to construct a DNN-based, multi-speaker acoustic model with speaker codes. 2) Next, we performed speaker adaptation by estimating code vectors for new speakers via backpropagation from a small amount of adaptation material. 3) Finally, we experimented with manually manipulating input code vectors to alter the gender and/or age characteristics of the synthesised speech. Experimental results show that high-performance multi-speaker models can be constructed using the proposed code vectors with a variety of encoding schemes, and that adaptation and manipulation can be performed effectively using the codes.

***Index Terms*—** Speech synthesis, DNNs, speaker adaptation, speech manipulation, voice morphing

1. INTRODUCTION

In some applications of speech technology, the flexibility and controllability of speech synthesis systems are important factors, in addition to the quality of the synthetic speech. It is well known that speech synthesis based on hidden Markov models (HMMs) is highly flexible and controllable. This is exemplified by 1) *speaker adaptation*, a technique to estimate a new acoustic model based on small amounts of speech data uttered by a new target speaker or in a new speaking style (e.g., a different emotion) [1, 2]; 2) *speaker interpolation*, a technique to morph together several speakers or emotions by interpolating mean vectors and variance matrices of Gaussian distributions from representative acoustic models [3]; and 3) *multiple regression HMMs* (MR-HMMs) [4], a technique that modifies the acoustic characteristics of synthesised speech by augmenting the text input with additional input values (control parameters).

Recently, speech synthesis using deep neural networks (DNNs) has become an active area of research, with applications to acoustic modelling [5, 6], duration modelling [7], feature extraction [8], and text analysis [9] having been investigated by various groups. It has been reported that DNN-based techniques have improved the quality of synthetic speech significantly; cf. [10]. A new DNN baseline system [11] added to the Blizzard Challenge 2016[1] also turned out to be significantly better than the standard HMM-based baseline (that uses a toolkit called HTS [12]), again confirming the speech quality improvements brought on by deep learning approaches.

To harness these quality improvements also in scenarios where a flexible synthesiser is required, a variety of speaker adaptation techniques have been proposed for DNN-based acoustic models. Wu et al. [13] proposed speaker adaptation using i-vectors as input, or by adapting hidden unit contributions (LHUC [14]), or by applying output transforms defined by GMMs, or combinations of these. Fan, et al. [15] assumed that the output layer in the DNN captures most speaker differences, and considered estimating speaker-dependent output layers using multi-speaker data, while keeping the hidden network layers shared across all speakers. This also allowed the model to be adapted for new speakers by only updating the regression layer [15]. However, their experiments only used four different speakers, with a relatively large amount of data (one hour) from each.

There have also been attempts to enable control of DNNs similar to multiple regression HMMs. In [16], two-dimensional per-sentence control-vector inputs to a DNN synthesiser were learned in an unsupervised fashion from a corpus of expressive speech. It was found that one direction in the (unlabelled) control-vector space had a consistent and interpretable influence on the generated speech, but the orthogonal direction did not.

In this paper, we consider augmenting the conventional text-based inputs of DNN-based acoustic models with auxiliary input features – collectively referred to as *input codes* – that include speaker codes as well as encodings of other labelled attributes, such as gender and age. We show that these input codes have capabilities that unify previous approaches to speaker adaptation, speaker interpolation, and multiple regression. More specifically, we first show that we can train a useful multi-speaker model based on input codes, even from speech database with over 100 speakers, each contributing only about 100 utterances. The model is trained exactly like a speaker-dependent acoustic model, but augmenting the linguistic inputs with additional features encoding each speaker's identity, gender, and age. Next, we show that the new model can perform speaker adaptation from small amounts of speech by estimating speaker codes and/or other input codes using backpropagation. Finally, we show that we can achieve speaker, gender, and age morphing by gradually changing the input code from one value to another.

We also investigate several ways of representing speaker, gender, and age features in vector form. We compare three different encodings, namely the standard one-hot vector form, random vectors,

*The first author performed this work while on an internship at the National Institute of Informatics, Japan, in 2016.

†The work presented in this paper was partially supported by EPSRC EP/J002526/1 (CAF), by the Core Research for Evolutional Science and Technology (CREST) from the Japan Science and Technology Agency (JST) (uDialogue project), by MEXT KAKENHI Grant Numbers (26280066, 26540092, 15H01686, 15K12071, 16H06302, 16K16096), and by The Telecommunications Advancement Foundation Grant.

[1]http://www.synsig.org/index.php/Blizzard_Challenge_2016_Workshop

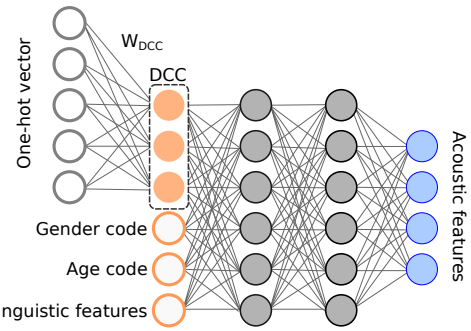


Fig. 1. Schematic representation of discriminant codes. W_{DCC} is a projection matrix that reduces one-hot vectors to DCC vectors.

and a data-driven numerical encoding sometimes called "discriminant condition codes" (DCCs) [17]. We further analyse the dimensionality required for the input code representations.

In the remainder of the paper, Section 2 describes training a neural network modelling several training speakers. Sections 3 and 4 explain speaker code estimation for new speakers via backpropagation and the modification of input codes to change the speaker, gender, and age characteristics of output speech. Section 5 evaluates the approaches in a number of experiments, while Section 6 concludes.

2. MULTI-SPEAKER MODELS USING INPUT CODES

We can train a multi-speaker acoustic model like a standard speaker-dependent model by simply pooling the data in a multi-speaker speech database. However, in order to retain speaker voice characteristics and to allow speaker adaptation later on, we augment the standard input features with auxiliary features such as the speakers identity, gender, and age. These input codes allow the DNN to discriminate individual speakers, gender, and age bands during training and synthesis; by inputting the average value of the input code, a kind of average voice is produced. This architecture is trivial, and similar ideas have cropped up in prior literature, not only in speech synthesis [13, 18, 19], but also in other fields [17, 20, 21, 22].

We first address the question of how to represent auxiliary features like speaker, gender, and age in vector form. In particular, we compare three different encodings, namely standard one-hot vectors, random vectors, and DCC vectors, for the speaker codes:

One-hot vector speaker codes: The standard one-hot vector speaker code is defined as follows: If there are N speakers in the training set, the *one-hot vector* (sometimes called the *1-of-k vector*) for the ith speaker is $\boldsymbol{s}_i = (s_1, s_2, ..., s_N)$ where each value s_n is 0 when $n \neq i$ and 1 when $n = i$.

Random-vector speaker codes: Speaker codes based on random vectors are vectors of a predetermined dimension K. The vector for speaker i is $\boldsymbol{s}_i = (s_1, s_2, ..., s_K)$ where s_k are random values, here sampled from a uniform distribution on $[0, 1)$. This scheme provides a simple method to reduce or expand the size of the speaker code.

Discriminant condition codes: DCCs were initially introduced by Xue et al. [17] for speech recognition. Figure 1 provides a simplified overview of the main idea. As seen in the figure, discriminant codes constitute hidden-unit activations obtained by projecting one-hot speaker codes into K dimensions using a matrix $W_{\mathrm{DCC}} \in \mathbb{R}^{N \times K}$. The DCCs are trained jointly with the weights and biases of a DNN [17, 20]. This code is equivalent to the one proposed by Watts et al. [16], except that the latter work considers synthesis and used DCCs to distinguish sentences instead of speakers.

2.1. Gender and age codes

For the gender and age codes, we consider two different representations: 1) one-dimensional numerical or binary representations, or 2) 1-of-k categorical representations. Gender information, for instance, may be represented by binary values, e.g., 0 for female and 1 for male. Alternatively, we may use a two-dimensional 1-of-k categorical representation where, e.g., the first dimension corresponds to male and the second dimension corresponds to female. For the age information, we may use the raw age as a one-dimensional numerical value, or we may divide speakers into several age bands and assign a similar 1-of-k categorical representation.

3. ADAPTING DNN ACOUSTIC MODELS USING INPUT CODES

To adapt the above multi-speaker models to a new speaker, one wants to find the speaker code that makes generated speech resemble the target speaker the most. Following [23], we use backpropagation (BP) to minimise the mean square prediction error over a small amount of data uttered by the target speaker.[2] This differs from [13, 24], whose adapted speaker codes do not minimise prediction error directly. Note that the BP algorithm only updates the speaker codes, without changing the DNN weights, in contrast to [19], which used fixed codes but added new weights. We use a well-trained multi-speaker acoustic model as described Section 2, initialised from the average speaker code, since average voice models are known to provide good starting points for HMM-based speaker adaptation [1]. Forward propagation and backpropagation are then iterated to estimate the new speaker code until a stopping criterion is satisfied.

4. MANIPULATING SPEECH USING INPUT CODES

Models from Section 2 can also be used to manipulate the acoustic characteristics of synthesised speech, achieving behaviour similar to HMM-based speaker interpolation as in [3]. We can linearly interpolate DCCs or other input codes, gradually changing them from the value for one speaker to that of another. This should perform speaker morphing.[3] Having labelled control parameters (here age and gender) in a synthesiser also opens up the possibility of manipulating these characteristics in a given voice, by changing these control parameters while keeping the speaker code fixed. This is similar to what multiple regression HMM approaches can achieve, and allows us to manipulate the gender and/or age of a given voice. Related regression techniques have previously been used to change perceived voice age in GMM-based singing voice conversion [25].

5. EXPERIMENTS

5.1. Experimental conditions

For our experiments, we used a corpus of studio-quality native Japanese speech uttered by 65 males and 70 females aged between 10 and 89. Of these, speech from 56 males and 56 females was used to train the multi-speaker DNN, with the remaining speakers (9 males and 14 females) held out for speaker adaptation, as listed in Table 1. The training-set speakers were chosen to be equally distributed for each age band (8 speakers for each age band and gender).

[2] The same technique may also be used to estimate age and gender codes, both to adapt synthesisers and to infer speaker age/gender. However, space limitations prevented us from exploring this possibility in the current paper.

[3] An infinite supply of novel voices can also be created by randomising new speaker codes, although this has been left for future work.

Table 1. Number of speakers used for training and adapting multi-speaker DNN acoustic models.

(a) Multi-speaker task				(b) Adaptation task			
Age	Male	Female	Total	Age	Male	Female	Total
10–20	8	8	16	10–20	0	2	2
21–30	8	8	16	21–30	2	2	4
31–40	8	8	16	31–40	2	2	4
41–50	8	8	16	41–50	1	2	3
51–60	8	8	16	51–60	2	2	4
61–70	8	8	16	61–70	2	2	4
71–	8	8	16	71–	0	2	2
Total	56	56	112	Total	9	14	23

Table 2. Models trained for the multi-speaker and adaptation tasks.

	Speaker code (S)		Gender code (G)		Age code (A)	
Model label	Type	Size	Type	Size	Type	Size
ONE-S	One-hot	112	N/A	N/A	N/A	N/A
ONE-SGA'	One-hot	112	One-hot	2	One-hot	7
ONE-SGA	One-hot	112	Numeric	1	Numeric	1
RND112-SGA	Random	112	Numeric	1	Numeric	1
RND008-SGA	Random	8	Numeric	1	Numeric	1
DCC112-SGA	DCC	112	Numeric	1	Numeric	1
DCC008-SGA	DCC	8	Numeric	1	Numeric	1

With approximately 100 utterances for each speaker, this yielded a total of 11,170 training-data utterances.

For the adaptation experiments, we used 100 utterances as adaptation material from each of 23 speakers not included in the training set. In this case, we were not able to balance all factors due to the limited number of speakers in the corpus, though we made our best efforts to make it as balanced as possible.

To evaluate the multi-speaker speech synthesis and speaker adaptation tasks, ten additional utterances per speaker were held out as test data for every speaker in the training and adaptation datasets.

The speech signal waveforms were sampled at 48 kHz, 16 bits. STRAIGHT [26] analysis was used to obtain 259-dimensional acoustic feature vectors every 5 ms, each comprising 60 mel-cepstral coefficients (with a bilinear frequency warping parameter of 0.77), a linearly interpolated fundamental frequency in the mel scale, and 25-dimensional band aperiodicities, along with their delta and delta-delta counterparts. The 259th feature was a binary voiced/unvoiced flag. During synthesis, the static and dynamic features were combined as described in [27] to produce the most likely speech trajectory sequence, based on a forced-alignment against the held-out natural speech (i.e., an oracle duration model was used).

For the input linguistic features, Open JTalk[4] was used to perform standard analysis of Japanese text for synthesis, including grapheme-to-phoneme conversion, part-of-speech tagging, and morphological analysis with the MeCab parser (based on [28]). The derived quin-phone identity and syntactic and prosodic text information were encoded into a 389-dimensional mixed discrete-and-numeric vector of linguistic features. This input vector was then augmented with the three types of auxiliary features mentioned earlier and used as the input to the neural network speech synthesiser.

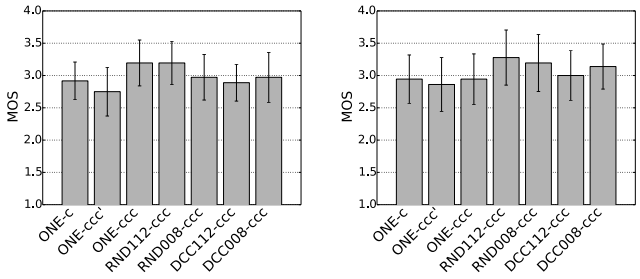
All acoustic models were feedforward DNNs with five hidden layers of 1024 nodes each. Sigmoid activation functions were used for all units in the hidden and output layers. The models were initialised randomly and trained to minimise mean square error using stochastic gradient descent for 10 epochs with the learning rate fixed at 0.05 and the minibatch size set to 256. The same learning rate was also used for learning discriminant codes.

When adapting to new speakers, speaker codes were estimated one-by-one for each of the 23 held-out speakers, using 10 BP epochs with learning rate 0.2. The estimated speaker code with the lowest error was used as the code for the new speaker in the experiments. Because of the small amount of adaptation data available, it was not obvious what would be the best training scheme. Our choice was computationally fast and appeared effective in our evaluation, but better optimisation procedures can probably be devised.

[4] http://open-jtalk.sourceforge.net/

Table 3. Objective evaluation results. Mel-cepstral distortion (MCD) was measured in dB and F_0 RMSE in Hz.

(a) Multi-speaker task			(b) Adaptation task		
Strategy	MCD	F_0 RMSE	Strategy	MCD	F_0 RMSE
ONE-a	7.64	52.06	ONE-a	7.49	53.90
ONE-c	5.62	23.79	ONE-e	6.02	23.71
ONE-ccc'	5.61	23.70	ONE-ecc'	6.06	23.75
ONE-ccc	5.58	23.43	ONE-ecc	6.01	23.54
RND112-ccc	5.60	23.48	RND112-ecc	6.46	24.98
RND008-ccc	5.60	23.06	RND008-ecc	6.49	29.44
DCC112-ccc	5.60	23.18	DCC112-ecc	6.03	24.77
DCC008-ccc	5.62	22.88	DCC008-ecc	6.43	27.18



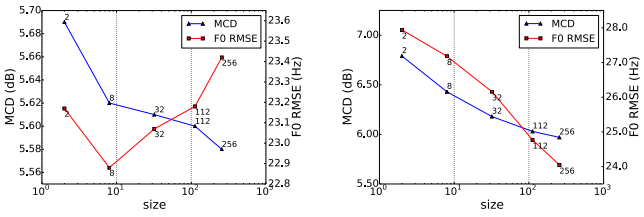
(a) Quality (b) Speaker similarity

Fig. 2. Subjective test results for the multi-speaker synthesis task.

A total of seven different acoustic models, summarised in Table 2, was trained and evaluated both objectively and subjectively. The three speaker-code variations described in Section 2 were used to train several models and compare their relative performance. Two representations of gender and age codes were also evaluated: numeric scalars and one-hot vectors. Numeric gender codes had a value of 0 or 1 (female or male, respectively) while numeric age-code values were set at the midpoint of each age interval (i.e., 15 through 65, also using 75 for speakers aged 70 and above). The categorical representations, meanwhile, used one-hot vectors with two and seven elements to encode gender and age categories, in that order.

Each system was assigned a label where the first part indicates the type of speaker code used (ONE, RND, or DCC) and, where relevant, its dimensionality K. The second part of the name indicates the auxiliary features included, with S standing for speaker code, G for gender code and A for age code. For example, ONE-S is the one-hot model trained without the gender and age codes, while ONE-SGA' is the model trained with these codes in their one-hot encoding; all other models used the numerical encoding of gender and age.

For our objective evaluations we considered the mel-cepstral distortion (MCD) and the F_0 root mean square error (RMSE). Two subjective tests were also conducted using nine paid, native Japanese listeners in a quiet office over high-end headphones: one to evaluate the multi-speaker task, and another for the adaptation task. In the former, listeners scored synthetic speech in terms of speech quality and speaker similarity to a reference speaker on a standard 5-point MOS scale. Four random samples were scored for each listener, system, and metric. To evaluate speaker adaptation performance, an AB test was used, where listeners chose which of two random output sentences sounded more similar to a reference recording of the target speaker, six times for each listener and system pair compared.



(a) Multi-speaker task (b) Adaptation task

Fig. 3. Objective evaluation of different discriminant code sizes.

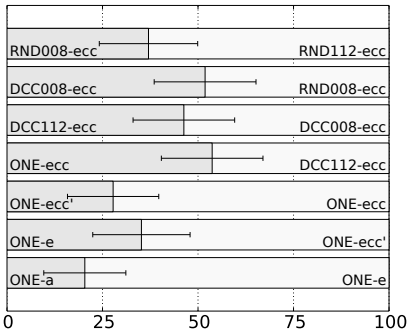


Fig. 4. AB-test results with 95% confidence intervals for the adaptation task. The x-axis indicates the observed relative preference (in %) for the system on the left-hand side in each pairwise comparison.

5.2. Evaluation of the multi-speaker task

Table 3 (a) lists our objective measurements for the multi-speaker speech synthesis task. Each system is labelled as follows: The first part shows which model was used while the second part indicates the value assigned to each global feature (S, G, and A, respectively) with 'a' denoting an average value while 'c' means that the corresponding feature was set to the correct value for the speaker in question.

The first system in Table 3 (a), ONE-a, shows the objective performance of the model ONE-S when all one-hot vector elements were assigned their average value. We view this as the average, reference voice of our system. Notably, this average voice was significantly less accurate than all other systems, showing that the multi-speaker system overall was successful at approximating the many different speakers in the corpus. In general, the multi-speaker systems showed relatively small objective differences, though we note that the F_0 RMSE slightly improved for both RND and DCC when reducing the speaker code (S) dimension to 8. The effect of the DCC size on objective measures is explored in Figure 3 (a). Interestingly, while the mel-cepstral distortion steadily decreases as the DCC size increases, the F_0 RMSE is the lowest when the DCC size is 8.

Subjective test results for quality and speaker similarity, with 95% confidence intervals based on Student's t-distribution, are plotted in Figures 2 (a) and (b), respectively. All models achieved similar scores with no statistically significant differences in either test. However, since ONE-ccc' got the poorest scores for both quality and similarity, we view the numerical encodings of age and gender as preferred, especially since those also are easier to manipulate.

5.3. Evaluation of the adaptation task

To evaluate the adaptation task, we used a similar objective evaluation as for the multi-speaker task. We also employed the same labelling convention, with one addition: the letter 'e' indicates a feature that was estimated using the method proposed in Section 3. The results are summarised in Table 3 (b). As mentioned earlier, only the speaker code was estimated in these experiments.

Like before, the first system in the table, ONE-a, is an average, reference voice based on ONE-S. It can be seen that all seven models, and thus all three types of speaker code, achieved significantly lower errors than the average voice by adapting to the 23 different target speakers. Compared to the known voices in Table 3 (a), we see that the performance was slightly worse on unknown voices, which is not unexpected. For both RND and DCC we note that the larger code dimensionality we tested achieved better objective results. Figure 3 (b) shows that, unlike the multi-speaker task, both mel-cepstral distortion and F_0 RMSE decreased as the dimensionality of the discriminant code used for adaptation increased.

Figure 4 shows the results of our AB tests comparing different adaptation schemes. Contrasting the average voice ONE-a against the adapted ONE-e (bottom test), listeners overwhelmingly preferred the adapted voice, confirming the efficacy of the adaptation. The next two tests from the bottom compared three one-hot speaker-code models with or without gender and age codes. The pattern is clear: adding age and gender information to the input improved the subjective adaptation performance, as expected, with the numerical encoding of gender and age being superior to the categorical encoding.

Several tests did not exhibit any statistically significant differences. In particular, 1) ONE and DCC performed similarly, 2) reducing DCC dimensionality from 112 to 8 did not hurt performance, and 3) RND performed similarly to DCC when using 8-dimensional speaker codes. Surprisingly, however, increasing RND dimensionality from 8 to 112 yielded a significant subjective improvement.

5.4. Manipulating speech characteristics

Next we performed three types of manipulation of the input codes. The first manipulation was speaker interpolation, where we gradually interpolated between the speaker codes of two speakers. We also experimented with changing the gender or age codes while keeping the other input codes constant; example manipulations can be found at `http://www.hieuthi.com/papers/icassp2017/`.

Although it is not straightforward to evaluate these manipulations because correct references are not available, informally, we found that changing the inputs altered the perception of the voice in a plausible manner for gender conversion, including some changes to the pitch. Similar to [25], modifying speaker age inputs also produced plausible audible differences, at least for larger changes. Manipulation effects were generally more pronounced when RND and DCC speaker codes were of lower dimensionality, with results appearing to be most consistent for the model DCC008-SGA.

6. CONCLUSIONS

In this paper, we considered augmenting the conventional text-derived inputs of DNN-based speech synthesis acoustic models with various input codes. We showed that a single DNN with input codes simultaneously was able to 1) learn a useful multi-speaker model even from a speech database with over 100 speakers and only a limited amount of speech from each; 2) perform adaptation to novel speakers; and 3) meaningfully control the synthetic speech output. This unifies previous approaches to adaptation and control in DNN-based synthesis. We also found that using gender and age codes improved speaker adaptation performance. Several speaker-code variations were investigated, finding that the most appropriate encoding and code size may depend on the task and the acoustic feature(s) considered. As future work, it would be interesting to consider similar input code setups for BLSTM-based acoustic models [29, 24].

7. REFERENCES

[1] Junichi Yamagishi and Takao Kobayashi, "Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training," *IEICE T. Inf. Syst.*, vol. 90, no. 2, pp. 533–543, 2007.

[2] Nose Takashi, Junichi Yamagishi, Takashi Masuko, and Takao Kobayashi, "A style control technique for HMM-based expressive speech synthesis," *IEICE T. Inf. Syst.*, vol. 90, no. 9, pp. 1406–1413, 2007.

[3] Makoto Tachibana, Junichi Yamagishi, Takashi Masuko, and Takao Kobayashi, "Speech synthesis with various emotional expressions and speaking styles by style interpolation and morphing," *IEICE T. Inf. Syst.*, vol. 88, no. 11, pp. 2484–2491, 2005.

[4] Nose Takashi, Makoto Tachibana, and Takao Kobayashi, "HMM-based style control for expressive speech synthesis with arbitrary speaker's voice using model adaptation," *IEICE T. Inf. Syst.*, vol. 92, no. 3, pp. 489–497, 2009.

[5] Heiga Zen, Andrew Senior, and Mike Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Proc. ICASSP*, 2013, pp. 7962–7966.

[6] Zhen-Hua Ling, Shi-Yin Kang, Heiga Zen, Andrew Senior, Mike Schuster, Xiao-Jun Qian, Helen M. Meng, and Li Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends," *IEEE Signal Proc. Mag.*, vol. 32, no. 3, pp. 35–52, 2015.

[7] Gustav Eje Henter, Srikanth Ronanki, Oliver Watts, Mirjam Wester, Zhizheng Wu, and Simon King, "Robust TTS duration modelling using DNNs," in *Proc. ICASSP*, 2016, pp. 5130–5134.

[8] Shinji Takaki and Junichi Yamagishi, "A deep auto-encoder based low-dimensional feature extraction from FFT spectral envelopes for statistical parametric speech synthesis," in *Proc. ICASSP*, 2016, pp. 5535–5539.

[9] Xin Wang, Shinji Takaki, and Junichi Yamagishi, "Enhance the word vector with prosodic information for the recurrent neural network based TTS system," in *Proc. Interspeech*, 2016.

[10] Oliver Watts, Gustav Eje Henter, Thomas Merritt, Zhizheng Wu, and Simon King, "From HMMs to DNNs: where do the improvements come from?," in *Proc. ICASSP*, 2016, pp. 5505–5509.

[11] Zhizheng Wu, Oliver Watts, and Simon King, "Merlin: An open source neural network speech synthesis system," in *Proc. SSW*, 2016.

[12] Heiga Zen, Takashi Nose, Junichi Yamagishi, Shinji Sako, Takashi Masuko, Alan W. Black, and Keiichi Tokuda, "The HMM-based speech synthesis system (HTS) version 2.0," in *Proc. SSW*, 2007, pp. 294–299.

[13] Zhizheng Wu, Pawel Swietojanski, Christophe Veaux, Steve Renals, and Simon King, "A study of speaker adaptation for DNN-based speech synthesis," in *Proc. Interspeech*, 2015.

[14] Pawel Swietojanski and Steve Renals, "Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models," in *Proc. SLT*, 2014, pp. 171–176.

[15] Yuchen Fan, Yao Qian, Frank K. Soong, and Lei He, "Multi-speaker modeling and speaker adaptation for DNN-based TTS synthesis," in *Proc. ICASSP*, 2015, pp. 4475–4479.

[16] Oliver Watts, Zhizheng Wu, and Simon King, "Sentence-level control vectors for deep neural network speech synthesis," in *Proc. Interspeech*, 2015, pp. 2217–2221.

[17] Shaofei Xue, Ossama Abdel-Hamid, Hui Jiang, Li-Rong Dai, and Qingfeng Liu, "Fast adaptation of deep neural network based on discriminant codes for speech recognition," *IEEE/ACM T. Audio Speech*, vol. 22, no. 12, pp. 1713–1725, 2014.

[18] Blaise Potard, Petr Motlicek, and David Imseng, "Preliminary work on speaker adaptation for DNN-based speech synthesis," Tech. Rep. Idiap-RR-02-2015, Idiap Research Institute, Martigny, Switzerland, 2015.

[19] Nobukatsu Hojo, Yusuke Ijima, and Hideyuki Mizuno, "An investigation of DNN-based speech synthesis using speaker codes," in *Proc. Interspeech*, 2016.

[20] Ossama Abdel-Hamid and Hui Jiang, "Fast speaker adaptation of hybrid NN/HMM model for speech recognition based on discriminative learning of speaker code," in *Proc. ICASSP*, 2013, pp. 7942–7946.

[21] Tian Tan, Yanmin Qian, Dong Yu, Souvik Kundu, Liang Lu, Khe Chai Sim, Xiong Xiao, and Yu Zhang, "Speaker-aware training of LSTM-RNNs for acoustic modelling," in *Proc. ICASSP*, 2016, pp. 5280–5284.

[22] Xie Chen, Tian Tan, Xunying Liu, Pierre Lanchantin, Moquan Wan, Mark J. F. Gales, and Philip C. Woodland, "Recurrent neural network language model adaptation for multi-genre broadcast speech recognition," in *Proc. Interspeech*, 2015, pp. 3511–3515.

[23] John S. Bridle and Stephen Cox, "RecNorm: Simultaneous normalisation and classification applied to speech recognition.," in *Proc. NIPS*, 1990, pp. 234–240.

[24] Yi Zhao, Daisuke Saito, and Nobuaki Minematsu, "Speaker representations for speaker adaptation in multiple speaker BLSTM-RNN-based speech synthesis," in *Proc. Interspeech*, 2016.

[25] Kazuhiro Kobayashi, Tomoki Toda, Tomoyasu Nakano, Masataka Goto, Graham Neubig, Sakriani Sakti, and Satoshi Nakamura, "Regression approaches to perceptual age control in singing voice conversion," in *Proc. ICASSP*, 2014, pp. 7904–7908.

[26] Hideki Kawahara, "STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds," *Acoust. Sci. Technol.*, vol. 27, no. 6, pp. 349–353, 2006.

[27] Keiichi Tokuda, Takayoshi Yoshimura, Takashi Masuko, Takao Kobayashi, and Tadashi Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. ICASSP*, 2000, pp. 1315–1318.

[28] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto, "Applying conditional random fields to Japanese morphological analysis," in *Proc. EMNLP*, 2004, pp. 230–237.

[29] Yuchen Fan, Yao Qian, Feng-Long Xie, and Frank K. Soong, "TTS synthesis with bidirectional LSTM based recurrent neural networks.," in *Proc. Interspeech*, 2014, pp. 1964–1968.