SPEECH WAVEFORM SYNTHESIS FROM MFCC SEQUENCES WITH GENERATIVE ADVERSARIAL NETWORKS

Lauri Juvela[1], *Bajibabu Bollepalli*[1], *Xin Wang*[2], *Hirokazu Kameoka*[3],
Manu Airaksinen[1], *Junichi Yamagishi*[2], *Paavo Alku*[1]

[1] Aalto University, Finland
[2] National Institute of Informatics, Japan
[3] NTT Communication Science Laboratories, NTT Corporation, Japan

ABSTRACT

This paper proposes a method for generating speech from filterbank mel frequency cepstral coefficients (MFCC), which are widely used in speech applications, such as ASR, but are generally considered unusable for speech synthesis. First, we predict fundamental frequency and voicing information from MFCCs with an autoregressive recurrent neural net. Second, the spectral envelope information contained in MFCCs is converted to all-pole filters, and a pitch-synchronous excitation model matched to these filters is trained. Finally, we introduce a generative adversarial network -based noise model to add a realistic high-frequency stochastic component to the modeled excitation signal. The results show that high quality speech reconstruction can be obtained, given only MFCC information at test time.

Index Terms— MFCC, Pitch prediction, Mel-filterbank inversion, Excitation modeling, Generative adversarial networks

1. INTRODUCTION

Mel freqency cepstral coefficients (MFCCs) [1] are widely used in speech applications, such as automatic speech recognition (ASR) [2] and speaker verification (ASV) [3, 4]. Since MFCCs are engineered for these tasks, their use discards lots of signal details that are considered irrelevant in the recognition task. The success of MFCCs in recognition and classification tasks is in part due to this lossy compression, which approximates perceptual properties in hearing [5]. Specifically, MFCCs separate spectral envelope from fine structure, and use a non-linear frequency resolution based on auditory scales.

Although MFCCs are usually considered sub-optimal for text-to-speech (TTS), and e.g. mel-generalized cepstrum (MGC) [6] is used instead to avoid utilizing filterbanks, reconstructing speech signals from a MFCC representation is sometimes needed. In ASR, for example, understanding causes behind recognition errors or analyzing effects of transcription errors might benefit from conversion of MFCCs to speech. Furthermore, state-of-the-art ASR and ASV systems utilizing MFCCs can give rise to novel transformation technologies, such as non-parallel voice conversion based on speaker verification models [7]. Despite being non-ideal for TTS, MFCCs constitute a state-of-the-art representation of speech information in most ASR and ASV systems and therefore conversion of this information to an audible speech waveform is justified.

While high-order MFCCs have been proposed for speech coding [8], the mel-filterbank sizes and discrete cosine transform (DCT) orders typically used in ASR and ASV result in the speech harmonic structure being smoothed out. In this case, the spectral information contained in MFCCs can be treated as an envelope. Given only this envelope, synthesis of speech requires pitch prediction, i.e. fundamental frequency (F0) and voicing information must be recovered from the MFCCs. This problem has been studied in a GMM-HMM framework [9, 10], where F0 and voicing were successfully predicted from a GMM joint distribution with MFCCs. As an obvious extension, modern sequence models, such as recurrent neural networks (RNNs), appear as potential tools for the task.

The method to convert MFCCs to speech proposed in [10] was relatively simplistic: the recovered spectral amplitude was assumed to be minimum phase and sampled at harmonic frequencies. In the time domain, this corresponds to exciting a minimum phase envelope filter with an impulse train. This process does not include any aperiodicity in synthesis of voiced speech, and loses the mixed phase characteristics in the excitation of natural speech (i.e. the glottal flow). Recently, neural net -based excitation models have been proposed to generate more realistic excitation waveforms for source-filter vocoding in statistical parametric speech synthesis (SPSS) [11]. Previous work used various acoustic features (such as F0, vocal tract and glottal source envelope parameters and harmonic-to-noise ratio), and trained a neural network to map them into a pitch-locked glottal excitation waveform. Unfortunately, this type of excitation models are limited due to the point-wise regression in the time domain, which results in smoothing and loss of high frequencies. To overcome this problem, a generative adversarial network (GAN) -based excitation model has been proposed recently [12]. However, GANs commonly suffer from training instability and mode collapse, which we propose to mitigate based on the ideas presented in [13] and [14].

In this paper, synthesis of speech from MFCCs is studied by presenting three main contributions: first, we show that F0 can be predicted from MFCCs with high accuracy by modifying a recent F0 model [15], originally proposed for SPSS. Second, we present an excitation model that maps MFCCs and F0 to excitation waveforms obtained by inverse filtering speech using an MFCC-derived envelope. Finally, we introduce an improved residual GAN-based noise model for generating the high-frequency stochastic component lost in the least-squares excitation model.

The results show that MFCCs can be used to synthesize speech with high quality, when no other information is available at test time. However, we do not advocate the use of MFCCs for envelope modeling in a purpose-built TTS system.The paper is structured as follows: section 2 overviews of the synthesis pipeline, section 2.1 describes the mapping of MFCCs to F0, section 2.2 details the MFCC to all-pole envelope conversion. Section 2.3 discusses the excitation pulse model and section 2.4 introduces a residual GAN noise model. Section 3 describes objective measures and listening experiment, and finally we conclude in section 4.

2. SYNTHESIS SYSTEM

An overview of the synthesis system is shown in Fig. 1. First, F0 is predicted from MFCCs as described in section 2.1. Then the MFCCs and predicted F0 is fed into the excitation pulse model detailed in section 2.3. The resulting smooth pulse is further fed into the residual GAN noise model (section 2.4). To create a continuous excitation signal, the generated pulses are joined pitch-synchronously [16], as determined by the generated F0. This excitation signal is finally filtered with the envelope reconstructed from the MFCC (section 2.2) to generate a speech waveform.

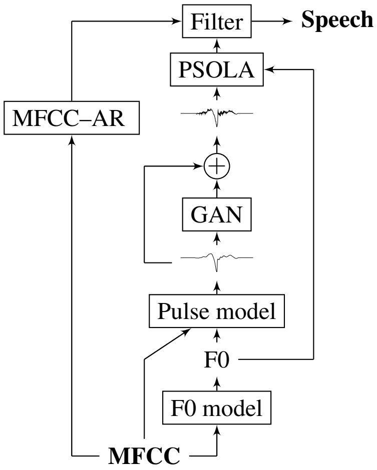

Fig. 1. System overview for MFCC-to-waveform synthesis.

2.1. F0 prediction model

The F0 model takes a sequence of MFCCs as input and generates the corresponding F0 track and voicing information from it. We use a variant of the recently proposed RNN-based model [15], which utilizes autoregressive output feedback links and hierarchical softmax for predicting quantized F0 classes from inputs. The F0 range is quantized linearly to 255 bins, and one additional class is reserved for unvoiced speech. In contrast to [15], we now use MFCCs instead of linguistic features as inputs, and only have feedback links at frame tier, as no linguistic tier information is available. Network parameters are listed in Table 1.

2.2. Envelope reconstruction from MFCC

This paper utilizes the widely used MFCC computation with HTK-style mel-filterbanks and DCT [17], as implemented in Librosa [18]. Spectrum-to-MFCC computation is composed of invertible pointwise operations and linear matrix operations that are pseudo-invertible in the least-squares sense. This leads to a straightforward reconstruction process: Let the MFCC sequence $\boldsymbol{C}$ be computed as

$$\boldsymbol{C} = \mathbf{D}\log(\mathbf{M}\boldsymbol{S}), \tag{1}$$

where $\boldsymbol{S}$ is a pre-emphasized STFT magnitude spectrogram, $\mathbf{M}$ is a mel-filterbank matrix, and $\mathbf{D}$ is a truncated discrete cosine transform matrix. The reconstruction of the magnitude spectrum is obtained simply by

$$\hat{\boldsymbol{S}} = \mathbf{M}^{+}\exp(\mathbf{D}^{+}\boldsymbol{C}), \tag{2}$$

where $\mathbf{D}^{+}$ is the pseudoinverse of $\mathbf{D}$ (which coincides with the classical zero-padding and inverse DCT procedure), and $\mathbf{M}^{+}$ is the pseudoinverse of $\mathbf{M}$. Unfortunately, the use of filterbank pseudoinverse does not guarantee non-negativity of the resulting spectrum, but this problem is mitigated by flooring the values to zero [19]. It is possible to instead obtain similar, but always non-negative reconstructions by using interpolation techniques (see e.g. [7, 10]), but we observe that the pseudoinverse behaves well in practice and gives envelopes with a sharper formant structure, compared to the interpolation methods.

An autoregressive all-pole model is fitted to $\hat{\boldsymbol{S}}$ in the conventional manner by computing an autocorrelation from the symmetrized square magnitude via IDFT, and then solving the resulting normal equations (see e.g. [20] for details). In this paper, we use 24 mel filters, 20:th order MFCCs, and 30:th order all-pole filters, at 16 kHz sample rate.

2.3. Excitation pulse model

Previously, an excitation model has been proposed for glottal vocoding in SPSS, by using a neural network that maps acoustic features to glottal excitation pulses [11]. A glottal source signal (differential volume flow through the vocal folds) is first obtained with glottal inverse filtering [21], after which excitation pulses are extracted by centering the excitation at a pitch mark, cosine windowing a two pitch period segment, and zero-padding the pulse to a fixed length. Finally before training, each acoustic feature frame is associated with a pulse at the nearest pitch mark.

A similar framework can be adopted generally in all source-filter model -based vocoding, where the filter allows inverse filtering the speech signal. In this paper, we use AR envelopes reconstructed from MFCCs to approximate the vocal tract filter, and otherwise treat the resulting excitation signals similarly to [11]. Furthermore, the model input acoustic features are now only MFCCs, log-F0 and voicing information. Reaper [22] is used to obtain the pitch marks, and only voiced frames are used to train the excitation model.

For the model architecture, we use a gated recurrent unit (GRU) layer at the input, since recurrent nets are powerful for encoding the acoustic sequence information. This is supported by previous research, where recurrent networks slightly improved excitation model performance in a TTS application [23]. Furthermore, convolution layers have been found convenient when working closer to the waveform level [12]. As result, we now use a GRU input, followed by a stack of 1D convolution layers, as listed in Table 1. However, a fundamental limitation in this kind of waveform modeling arises from the point-wise least-squares training criterion. The model will inevitably regress towards a conditional average, given the inputs, which leads to smoothed waveforms and loss of high frequencies. This is illustrated in Fig. 2.

2.4. Residual GAN model

Modeling the aperiodic component of voiced speech in the current synthesis system resembles a previous GAN glottal excitation model [12], but now the GAN is conditioned on a smooth generated pulse, and the model is forced to generate only an additive residual component. The training procedure combines LS-GAN [24] with GAN-based similarity metric learning [14]. Furthermore, the residual connections in the generator model are adopted from a GAN postfilter architecture [13]. Generator and discriminator architectures are shown in Figs. 3 and 4, respectively, with details listed in Table 1.

Generator input channels are a smooth excitation pulse waveform $\hat{x}$ given by the pulse regression DNN, and white Gaussian

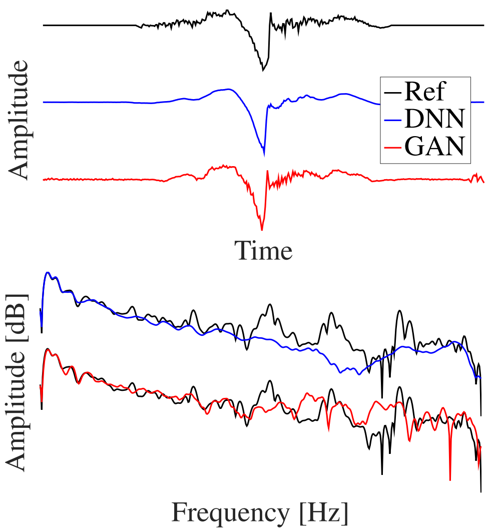

Fig. 2. Excitation pulses shown in the time and frequency domain. DNN pulse model (blue) fits the overall reference pulse (black) shape, but high-frequency stochastic components are smoothed out, whereas GAN (red) is able to generate realistic high-frequency components.

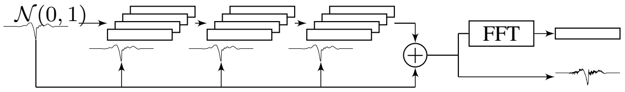

Fig. 3. Generator architecture. Two input channels contain a smooth excitation pulse and white noise, and the pulse is fed to every layer in a residual channel. Additivity at output forces the generator to learn a residual noise-like model.

noise of the same length. The smooth waveform is further fed into each layer as an additional residual channel. Explicit additivity of $\hat{x}$ and the convolution layers' output ensures that the output resembles residual noise. Finally, a FFT magnitude layer allows the training process to simultaneously see the output in the time and frequency domains, and the error signals can propagate via both routes.

Discriminator input channels see the signal both in the time and frequency domain. Strided convolutions are used to gradually reduce the convolution layer sizes, finally resulting in a single value for classifying the input sample as real of fake. In addition, the generator is allowed to peek into discriminator activations at layer L, and use the implied similarity metric to match a generated data mini-batch directly with the corresponding real data.

More formally, the discriminator D attempts to output a "real" value 1 for samples from real data distribution $x \sim p_{\mathrm{data}}(x)$, and a "fake" value 0 for samples $x' \sim p_G(z|\hat{x})$ coming from the generator G, where $\hat{x}$ is a smooth output of the previous pulse model, and z is sampled from a standard Gaussian distribution. Similarly to LS-GAN, the discriminator loss function to be minimized is

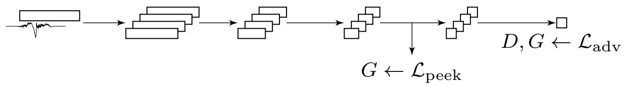
$D, G \leftarrow \mathcal{L}_{\mathrm{adv}}$
$G \leftarrow \mathcal{L}_{\mathrm{peek}}$

Fig. 4. Discriminator architecture. Input channels contain time and frequency domain views of the signal. Adversarial loss $\mathcal{L}_{\mathrm{adv}}$ is used to train both D and G. Additionally, G is allowed to peek into discriminator and use the loss $\mathcal{L}_{\mathrm{peek}}$ to match real and generated data activations.

$$\mathcal{L}_{\mathrm{adv}}(D) = \frac{1}{2}\,\mathbb{E}_x\left[(D(x)-1)^2\right] + \frac{1}{2}\,\mathbb{E}_{x'}\left[(D(x'))^2\right], \tag{3}$$

where $x' = G(z|\hat{x})$. Simultaneously, the generator is trained to fool the discriminator to produce the "real" value given generator output

$$\mathcal{L}_{\mathrm{adv}}(G) = \frac{1}{2}\,\mathbb{E}_{x'}\left[(D(x')-1)^2\right]. \tag{4}$$

To facilitate learning, the generator is allowed to peek into discriminator activations at layer L and attempt to match its generated output with target data.

$$\mathcal{L}_{\mathrm{peek}}(G) = \frac{1}{2}\,\mathbb{E}_{x,x'}\left[(D_L(x)-D_L(x'))^2\right]. \tag{5}$$

This resembles the "learned similarity metric" proposed in [14].

The training procedure alternates between minimizing $\mathcal{L}_{\mathrm{adv}}(D)$ for discriminator, and $\mathcal{L}_{\mathrm{adv}}(G) + \mathcal{L}_{\mathrm{peek}}(G)$ for generator. The discriminator is kept fixed while training the generator.

2.4.1. Fourier transform layer

In the GAN network, a non-trainable FFT layer is implemented to explicitly output the log spectral magnitude of the input, while allowing backpropagation of error gradients through the layer. The discrete Fourier transform consists of two differentiable linear operations, given by $F = F_R + iF_I$, where F_R and F_I are the cosine and sine basis matrices. Magnitude is obtained simply by point-wise squaring and summing of the real and imaginary part outputs. Applying logarithm and scaling pointwise is also differentiable. All of the operations are readily available in Theano, allowing easy integration to our computational graph.

3. EXPERIMENTS

3.1. Model training

The network parameters used are listed in Table 1. "Dense" denotes fully connected feedforward layers, and "BN" denotes batch normalization. The F0 model was trained with a modified version of CURRENNT [25], available online[1]. Excitation models use Keras [26] with Theano [27] backend (code available online[2]). One dimensional convolution layers are used throughout, and paramater K in Conv1D(K) denotes the number of channels in a layer.

The F0 and pulse regression DNN models were trained with the Adam optimizer [28] using early stopping, and frozen after training. Unfortunately, there is no established procedure for measuring convergence of GANs. After the first few epochs, the generator starts to produce visually plausible results, while the sound quality varies from epoch to epoch. In the end, we trained the GAN models for a total of 20 epochs, and chose the best model from the last five epochs by informal listening.

[1] https://github.com/TonyWangX/

[2] https://github.com/ljuvela/ResGAN

Table 1. Network parameters.

F0 model, *input*: MFCC(20)
Dense (256) × 2, tanh
BLSTM (128), tanh
LSTM (128), tanh, feedback link from output
SoftMax(256), 256 class quantization for F0 and voicing
output: F0(1), VUV(1)
Pulse model, *input*: MFCC(20), LF0(1), VUV(1)
GRU (50), ReLU, BN, context len = 40
Dense (400), ReLU, BN
Conv1D (100) × 4, LReLU, BN, width = 15
Conv1D (1), LReLU, BN, width = 15
output: Pulse(400)
GAN generator, *input*: Noise(400), Pulse(400)
Conv1D (100+1) × 3, LReLU, BN, width = 15
Conv1D (1), tanh, BN, width = 15
output: Pulse(400), FFT-of-Pulse(400)
GAN discriminator, *input*: Pulse(400), FFT-of-Pulse(400)
Conv1D (64), LReLU, BN, width=7, stride=3
Conv1D (128), LReLU, BN, width=7, stride=3
Conv1D (256), LReLU, BN, width=7, stride=3, *peek output*
Conv1D (128), LReLU, BN, width=5, stride=2
Conv1D (1), LReLU, BN, width=3, stride=2
output: Real/Fake classification (1)

3.2. Speech material

We trained two speaker-specific systems using existing SPSS training data. Both speakers are professional UK English voice talents, with ”Nick” (male) dataset comprising 2542 utterances, totaling 1.8 hours, and ”Jenny” (female) dataset comprising 4080 utterances, totaling approx. 4 hours. A randomly selected set of 100 utterances was kept for testing for both speakers, and the rest were used for training. 16 kHz sample rate was used throughout the study.

3.3. F0 objective measures

F0 model performance is measured by root-mean-squared error (RMSE) of voiced F0, voicing decision error percentage (VUV error), and correlation coefficient between reference and generated F0 values. Table 2 lists test set objective measures. An example of a generated F0 contour from ”Jenny” test set is shown in Fig. 5.

Table 2. Objective measures on F0.

	RMSE	VUV error	corr.
Jenny	9.3579	1.26 %	0.9939
Nick	4.2332	2.31 %	0.9969

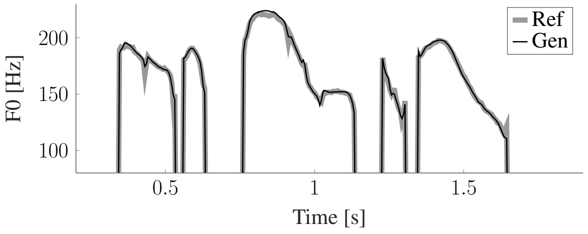

Fig. 5. Example of reference and generated F0 contours.

3.4. Listening test

Three systems were compared in a DMOS [29] listening test. All systems use the all-pole envelopes reconstructed from MFCCs, and F0 and voicing information generated by the F0 model. White noise was used for unvoiced excitation in all systems. System ”Impulse” uses a simple impulse train for voiced excitation. System ”DNN” uses the smooth excitation pulses generated by the DNN excitation model, and ”GAN” additionally uses the residual GAN noise model.

Natural speech signal was used as the reference and the listeners were asked to rate the degradation of the synthetic test sample from 1 (very annoying) to 5 (inaudible). The test was conducted on the CrowdFlower crowd sourcing platform [30], where it was made available in English speaking countries, and the top four countries in EF English Proficiency Index ranking [31]. Each test case was evaluated by 50 listeners on 15 test set utterances. Evaluation scores are shown in Fig. 6. Mode value is marked with a horizontal line and the mean value with a triangle. Box and whiskers show 25 and 75 percentile boundaries, respectively. A Mann–Whitney U-test, with correction for listener and utterance bias [32], found all differences between systems statistically significant. Audio samples are available online at http://tts.org.aalto.fi/mfcc_synthesis/.

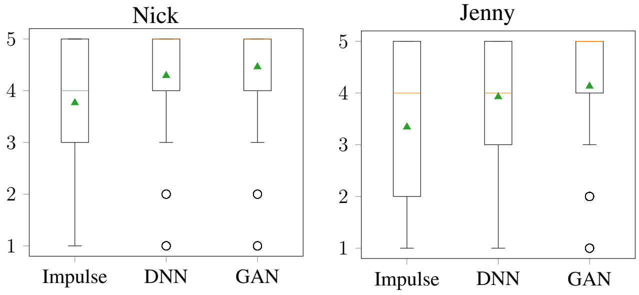

Fig. 6. DMOS listening test results.

4. CONCLUSION

This paper presented a method for speech reconstruction from MFCCs. F0 contours can be generated from MFCCs with high accuracy, using an autoregressive RNN operating on quantized F0 values. The spectral envelope information in MFCCs was recovered by least-squares inversion of the MFCC computation, and a DNN excitation model was trained for the MFCC-derived filters. Additionally, we proposed a residual GAN noise model that can be used to generate a realistic stochastic signal component without explicit parametrization of aperiodicity or similar features.

The listening tests show that a reasonable quality speech reconstruction is obtained already from the MFCC-derived envelope and impulse train excitation with the generated F0. Further improvements were gained with the proposed DNN exitation model and GAN noise model, resulting in high quality speech synthesis from the MFCCs.

5. ACKNOWLEDGEMENTS

This work was supported by the Academy of Finland (proj. no. 284671 and 312490.) and MEXT KAKENHI Grant Numbers (26280066, 15H01686, 15K12071, 16H06302). We acknowledge the computational resources provided by the Aalto Science-IT project.

6. REFERENCES

[1] Steven Davis and Paul Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[2] Lawrence R Rabiner and Biing-Hwang Juang, *Fundamentals of Speech Recognition*, PTR Prentice Hall, 1993.

[3] Tomi Kinnunen and Haizhou Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12–40, 2010.

[4] John HL Hansen and Taufiq Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 74–99, 2015.

[5] Hynek Hermansky, Jordan R Cohen, and Richard M Stern, "Perceptual properties of current speech recognition technology," *Proceedings of the IEEE*, vol. 101, no. 9, pp. 1968–1985, 2013.

[6] Keiichi Tokuda, Takao Kobayashi, Takashi Masuko, and Satoshi Imai, "Mel-generalized cepstral analysis – a unified approach to speech spectral estimation.," in *Proc. ICSLP*, 1994, pp. 18–22.

[7] Tomi Kinnunen, Lauri Juvela, Paavo Alku, and Junichi Yamagishi, "Non-parallel voice conversion using i-vector PLDA: Towards unifying speaker verification and transformation," in *Proc. ICASSP*, 2017, pp. 5535–5539.

[8] Laura E Boucheron, Phillip L De Leon, and Steven Sandoval, "Low bit-rate speech coding through quantization of mel-frequency cepstral coefficients," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 610–619, 2012.

[9] Xu Shao and Ben Milner, "Predicting fundamental frequency from mel-frequency cepstral coefficients to enable speech reconstruction," *The Journal of the Acoustical Society of America*, vol. 118, no. 2, pp. 1134–1143, 2005.

[10] Ben Milner and Xu Shao, "Prediction of fundamental frequency and voicing from mel-frequency cepstral coefficients for unconstrained speech reconstruction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 24–33, 2007.

[11] Lauri Juvela, Bajibabu Bollepalli, Manu Airaksinen, and Paavo Alku, "High-pitched excitation generation for glottal vocoding in statistical parametric speech synthesis using a deep neural network," in *Proc. ICASSP*, March 2016, pp. 5120–5124.

[12] Bajibabu Bollepalli, Lauri Juvela, and Paavo Alku, "Generative adversarial network-based glottal waveform model for statistical parametric speech synthesis," in *Proc. Interspeech*, 2017, pp. 3394–3398.

[13] Takuhiro Kaneko, Hirokazu Kameoka, Nobukatsu Hojo, Yusuke Ijima, Kaoru Hiramatsu, and Kunio Kashino, "Generative adversarial network-based postfilter for statistical parametric speech synthesis," in *Proc. ICASSP*, March 2017, pp. 4910–4914.

[14] Takuhiro Kaneko, Hirokazu Kameoka, Kaoru Hiramatsu, and Kunio Kashino, "Sequence-to-sequence voice conversion with similarity metric learned using generative adversarial networks," in *Proc. Interspeech*, 2017, pp. 1283–1287.

[15] Xin Wang, Shinji Takaki, and Junichi Yamagishi, "An RNN-based quantized F0 model with multi-tier feedback links for text-to-speech synthesis," in *Proc. Interspeech*, 2017, pp. 1059–1063.

[16] Eric Moulines and Jean Laroche, "Non-parametric techniques for pitch-scale and time-scale modification of speech," *Speech Communication*, vol. 16, no. 2, pp. 175–205, 1995.

[17] Steve Young, Gunnar Evermann, Mark Gales, Thomas Hain, Dan Kershaw, Xunying Liu, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, et al., *The HTK book*, vol. 3, Cambridge University, 2002.

[18] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto, "Librosa: Audio and music signal analysis in python," in *Proc. of the 14th Python in Science Conference*, 2015, pp. 18–25.

[19] Laura E Boucheron and Phillip L De Leon, "On the inversion of mel-frequency cepstral coefficients for speech enhancement applications," in *Proc. ICSES*. IEEE, 2008, pp. 485–488.

[20] John Makhoul, "Linear prediction: A tutorial review," *Proceedings of the IEEE*, vol. 63, no. 4, pp. 561–580, Apr 1975.

[21] Manu Airaksinen, Tuomo Raitio, Brad Story, and Paavo Alku, "Quasi closed phase glottal inverse filtering analysis with weighted linear prediction," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 3, pp. 596–607, March 2014.

[22] David Talkin, "REAPER: Robust Epoch And Pitch EstimatoR," https://github.com/google/REAPER, 2015.

[23] Lauri Juvela, Xin Wang, Shinji Takaki, Manu Airaksinen, Junichi Yamagishi, and Paavo Alku, "Using text and acoustic features in predicting glottal excitation waveforms for parametric speech synthesis with recurrent neural networks," in *"Proc. Interspeech"*, Sep. 2016, pp. 2283–2287.

[24] Xudong Mao, Qing Li, Haoran Xie, Raymond Y.K. Lau, and Zhen Wang, "Least squares generative adversarial networks," *arXiv preprint arXiv:1611.04076v2*, 2017.

[25] Felix Weninger, "Introducing CURRENNT: The Munich open-source CUDA recurrent neural network toolkit," *Journal of Machine Learning Research*, vol. 16, pp. 547–551, 2015.

[26] François Chollet et al., "Keras," https://github.com/fchollet/keras, 2015.

[27] Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, James Bergstra, Ian J. Goodfellow, Arnaud Bergeron, Nicolas Bouchard, and Yoshua Bengio, "Theano: new features and speed improvements," Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

[28] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.

[29] "Methods for Subjective Determination of Transmission Quality," Recommendation P.800, ITU-T SG12, Geneva, Switzerland, Aug. 1996.

[30] CrowdFlower Inc., "Crowd-sourcing platform," https://www.crowdflower.com/, Accessed: 2017-10-25.

[31] "EF English proficiency index," http://www.ef.com/epi/, Accessed: 2017-10-24.

[32] Andrew Rosenberg and Bhuvana Ramabhadran, "Bias and statistical significance in evaluating speech synthesis with mean opinion scores," in *Proc. Interspeech*, 2017, pp. 3976–3980.