SFNet: A Computationally Efficient Source Filter Model Based Neural Speech Synthesis

Achuth Rao MV, *Student Member, IEEE*, and Prasanta Kumar Ghosh, *Senior Member, IEEE*

***Abstract*—Recently, neural speech synthesizers have achieved a high-quality synthesis for text-to-speech applications, but a real-time synthesis is possible only in the devices which have high memory and allow large computational complexity. In this work, we reduce the complexity of a speech synthesizer by reformulating the source-filter model of speech where the excitation signal is modeled as a sum of two signals. The first signal contains an impulse train that is computed from the pitch sequence. The second signal is modeled as white noise passed through a filter bank with frequency dependent gains. The parameters of the reformulated source-filter model are predicted using a neural network, referred to as SFNet. The network parameters are learnt by training the network using l_1-error between the log Mel-spectrum of the predicted waveform and that of the ground-truth waveform. We demonstrate that there is a significant reduction in the memory and computational complexity compared to the state-of-the-art speaker independent neural speech synthesizer without any loss of the naturalness of the synthesized speech.**

***Index Terms*—Neural vocoder, source-filter model, computational complexity, Mel-spectrum.**

I. Introduction

MANY end-to-end text to speech synthesis systems convert letter sequences to speech features and use a vocoder to convert the speech features to the speech waveform [1]. The main limitation of conventional vocoders, such as Griffin-Lim [2] or WORLD [3], is that the errors due to the assumption made by vocoders are not corrected by the end-to-end network. In contrast, neural speech synthesizers convert these speech features to the speech waveform in a data-driven manner. Neural speech synthesis algorithms have achieved human level naturalness in text-to-speech synthesis [4], [5] and lowbit rate coding [6], but efficiently modeling the long term dependencies present in the waveform is still a challenging problem.

WaveNet is one of the first neural speech synthesis algorithms to achieve human level performance [7]. It uses a dilated convolution to model the long term dependencies and synthesizes the speech in an autoregressive (AR) manner. The computational complexity of the model is close to hundreds of billions of floating-point operations per second (GFLOPS). Various modifications of WaveNet have been proposed to reduce the computational complexity of the algorithm [8], by eliminating the sequential nature of the synthesis. However, the real time synthesis is possible only in graphical processing units (GPUs). There are two different kinds of approaches in the literature to reduce the complexity. In the first class of approaches, various structured neural networks are used to model the long term dependencies and reduce the time by using parallel nature of the model or reduced computation per sample. For example, WaveRNN uses a sparse recurrent network (RNN) and dual softmax function [9] to reduce the generation time for each sample. WaveGlow uses inverse-flow based generative models to synthesize speech [10]. In the second class of approaches, complexity is reduced by modeling the spectrum envelope using linear prediction (LP), but the excitation signal is generated using neural networks such as WaveNet [11], [12], sinusoidal model [13] and RNN with sparse weights [9]. Some of these models are also used for low bitrate coding, where the speech waveform is predicted from the quantized speech features [14]. However, the required computational complexity is high for devices which do not have powerful GPU/CPU and have limited power supply.

It is observed in the literature that the frame level phase is not important for the naturalness of the speech. For example, in the case of voiced speech, the exact phase of the impulse train in a synthesized speech need not match with that in the original speech. In the case of unvoiced sound, the exact realization of white noise (phase) need not match with the original speech to obtain a natural quality synthesized speech [3], [15]. The existing synthesizers such as LPCnet [16] try to model the exact phase of the excitation. In this paper, we exploit this fact to reduce the complexity of synthesis. The proposed synthesis network is referred to as SFNet, in which we reformulate the traditional source filter model, where we use an LP to model spectral envelope and propose to represent the excitation signal using two source signals. The first source signal represents the impulse train, primarily for the voiced speech, that is predicted from the pitch sequence. The second source signal represents the colored noise component present in both voiced and unvoiced speech. We propose to use a neural network to predict parameters of the source and the filter. The reformulated source-filter model is used to synthesize the speech signal from the predicted parameters.

II. Reformulation of Source-Filter Model for Speech

A speech frame is often represented by a source filter model [17], which assumes that the speech samples ($x[n]$) in that frame are obtained by passing an excitation signal through a filter. In the source filter model, the glottal excitation acts as a source and the

Manuscript received March 27, 2020; revised May 18, 2020; accepted June 15, 2020. Date of publication June 25, 2020; date of current version July 17, 2020. This work was supported by the Department of Science and Technology, Government of India. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Daniele Giacobello. *(Corresponding author: Achuth Rao MV.)*

The authors are with the Department of Electrical Engineering, Indian Institute of Science, Bangalore 560012, India (e-mail: achuthr@iisc.ac.in; prasantg@iisc.ac.in).

Digital Object Identifier 10.1109/LSP.2020.3005031

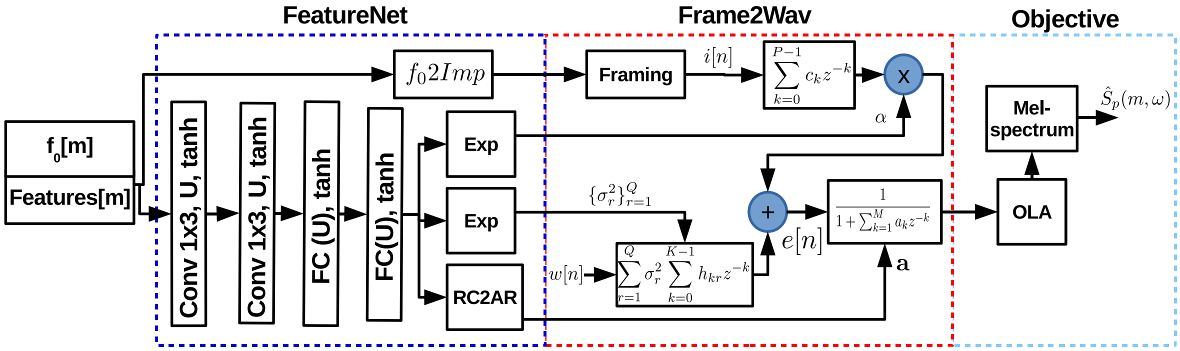

Fig. 1. Architecture of the proposed SFNet. The convolution (Conv), fully connected (FC) layer use $\tanh$ activation function, m denotes the m-th frame and Exp indicates a linear layer followed by an exponential function.

vocal tract and the lips together act as a filter. It is assumed that the vocal tract filter is an all-pole filter and lip-radiation is a first order all-zero filter. Based on the state of the glottis the speech can be classified into two categories of sounds – voiced sound and unvoiced sound. In the case of a voiced sound, the quasi-periodic vibration of the glottis is modeled using an impulse train convolved with a filter $\frac{1}{G(z)}$. In the literature, it has been reported that the voiced sound also has an aperiodic component, which is important for the naturalness of the speech [3]. Hence, we model the z-transform of the signal within a frame of voiced speech as follows: $X_v(z) = \frac{L(z)}{A(z)}(\frac{\alpha I(z)}{G(z)} + B(z)W(z))$, where $L(z)$ is the z-transform of the lip radiation filter, $\frac{1}{A(z)} = \frac{1}{1+\sum_{k=1}^{M} a_k z^{-k}}$ is the z-transform of an M-th order all-pole vocal tract filter, $I(z)$ is the z-transform of the unit impulse train ($i[n]$), and α is the strength of the impulse train in the glottal excitation. $B(z)$ is the z-transform of the noise filter and $W(z)$ is the z-transform of the white noise ($w[n]$) sampled from $\mathcal{N}(0,1)$ corresponding to the aperiodic component of the glottal excitation. For the unvoiced sound, the glottal excitation is assumed to be the colored noise ($W(z)B(z)$) and, hence, the z-transform of unvoiced speech signal is modeled as follows: $X_{unv}(z) = \frac{L(z)B(z)W(z)}{A(z)}$. We first pre-emphasize the voiced/unvoiced speech using the filter $P(z) = 1 - \kappa z^{-1}$. The pre-emphasized speech is given by $X_v^p(z) = \frac{1}{A(z)}(\frac{\alpha I(z)P(z)L(z)}{G(z)} + L(z)P(z)B(z)W(z))$ and $X_{unv}^p(z) = \frac{1}{A(z)} P(z)L(z)B(z)W(z)$. Thus, a speech frame can be represented using a source filter model as follows:

$$X^p(z) = \frac{1}{A(z)}\left(\alpha I(z)C(z) + W(z)D(z)\right) = \frac{E(z)}{A(z)} \quad (1)$$

where, $C(z) = \frac{L(z)P(z)}{G(z)}$ and $D(z) = P(z)L(z)B(z)$. $\alpha \neq 0$ for voiced speech, while $\alpha = 0$ for unvoiced speech. $E(z)$ represents the z-transform of the excitation signal $e[n]$.

In case of LPCnet [16] and GlotNet [11], $e[n]$ is predicted using a neural network and $A(z)$ is estimated using conventional LP. It is clear from eq. 1 that the excitation can be further broken down into simple operations. According to eq. 1, the $C(z)$ and $D(z)$ should be ARMA filters. However, in this work, we assume $C(z) = \sum_{k=0}^{P-1} c_k z^{-k}$ to be a fixed (across all frames) FIR filter of order P. The noise filter $D(z)$ models the noise in different subands. Hence, we pass the white noise through a filter bank with Q filters, each of length K ($H = \{h_{rk}, 0 \leq k \leq K-1, 1 \leq r \leq Q\}$). The colored noise output is constructed by varying the gain of each filter (σ_r^2) and, hence, $D(z) = \sum_{r=1}^{Q} \sigma_r^2 \sum_{k=0}^{K-1} h_{rk} z^{-k}$. The parameters of the models for each frame are $\theta = \{\mathbf{a}, \{\sigma_r^2\}_{r=1}^{Q}, \alpha\}$ and a set of common parameters across frames $\zeta = \{H, \mathbf{c}\}$, where $\mathbf{c} = [c_0, c_2, \ldots c_{P-1}]$ and $\mathbf{a} = [a_1, a_2, \ldots, a_M]$. We propose a data-driven neural network architecture, called SFNet, to predict the parameters (θ) of the source and the filter.

III. Proposed SFNet

The network architecture of the SFNet is shown in Fig. 1. The SFNet has three main parts– (1) FeatureNet: It operates at frame level and predicts the frame-level parameters (θ) from the input features. In this work, we use 18 dimensional mel-cepstrum, pitch correlation and pitch value as inputs [14]. For low bit rate applications, the features would be quantized [14]. However, for text-to-speech synthesis, they would be computed from the text using another neural network. In this work, we compute the features using a frame size of 20ms with 10ms overlap. (2) Frame2Wav: It operates at the sample rate and converts frame level filter parameters to waveform. (3) Objective function: This is the loss function used to train the network. The details of every part are described below.

A. FeatureNet

Given a batch of features from consecutive frames, we predict the model parameters (θ). The features are passed through two convolution layers each having U filters of length three. This results in a context of 5 frames (2 after and 2 before). The learnt representation is passed through two fully connected layers with U-units to get an intermediate representation. This architecture is similar to the LPCnet [16]. We use another linear layer with different activation functions for different model parameters as the range of values for various model parameters are different. The α and $\{\sigma_r^2\}_{r=1}^{Q}$ parameters take only non-negative values. Hence, we use a linear layer followed by an exponential function and it is indicated by Exp. A direct prediction of AR filter coefficients does not guarantee the stability of the filter. Hence we predict the reflection coefficients (RC) from the intermediate representation. To make sure that the predicted filter coefficients result in a stable filter, we constrain RC values between $+1$ and -1 using tanh activation. The RC values are converted to LP coefficients using Levinson recursion [18]. As the α, $\{\sigma_r^2\}_{r=1}^{Q}$ and $\mathbf{a}$ do not span the entire parameter space, we assume that $U < Q + M + 1$ and the value of U is experimentally determined.

B. Frame2Wave

Given the model parameters, predicted from the FeatureNet, the excitation is constructed and filtered to get the speech waveform. Given the sequence of pitch values, we construct the impulse train ($i[n]$). The pitch value $f_0[m]$ is interpolated to each sample using linear interpolation to get $f_0^{int}[n]$. The instantaneous phase is computed as $\phi[k] = \frac{2\pi}{fs} \sum_{m=0}^{k} f_0^{int}[m]$, where f_s is the sampling frequency. The discontinuity in $\phi[k] \bmod 2\pi$ will be the location of the unit impulses. This impulse train of unit strength is generated in an online manner. This block is indicated by $f_0 2Imp$ in the block diagram in Fig. 1. It is to be noted that the initial phase of the impulse is unknown and, hence, the impulse locations do not coincide with the ground truth locations. We ignore the jitter/shimmer present within a frame. We frame the impulse train of an entire utterance to get the excitation for each frame ($i[n]$). This block is indicated by Framing in the block diagram in Fig. 1. Given the model parameters $\{\mathbf{a}, \{\sigma_r^2\}_{r=1}^{Q}, \alpha\}$, predicted by the network and $i[n]$, $w[n]$, for each frame, is sampled from $\mathcal{N}(0, 1)$. Using these, the speech frame is synthesized using eq. 1. We use frame level waveform prediction and overlap add (OLA) using Hanning window to synthesize the final waveform.

C. Objective Function

The neural vocoders use the cross entropy with the quantized output as the objective function. As the proposed method ignores the frame level phase of the excitation, we first convert the generated waveform to frame level Mel-spectrum so as to make the objective function insensitive to the frame level phase. We use a window size of 20ms with 10ms overlap to compute the Mel-spectrum. Similar to [1], we use l_1 loss between the ground truth log Mel-spectrum and the generated log Mel-spectrum to train the network in an end-to-end manner[1]. The OLA is included in the objective function to help the network compensate for the errors because of OLA.

IV. Experiments and Results

A. Database and Baseline Schemes

The TSP Speech Database [19] and IIIT-H Indic Speech Database [20] are used for experiments. The TSP Speech Database consists of approximately 4 hours of data (at a sampling rate of 16 kHz) from a total of 1400 utterances spoken by 24 speakers (12 male, 12 female). The IIIT-H Indic Speech Database consists of recordings in six different languages with one speaker per language. In this work we select only Hindi and Kannada language speakers.

We use the LPCnet as the baseline scheme, a neural vocoder that is speaker independent which has been shown to achieve a good naturalness with low complexity [16]. The main component of the LPCnet is the two GRU layers. Here we choose a size of 192 units for the first GRU layer and 16 units for the second GRU for the comparison and indicate this scheme by LPCnet-192 [21].

B. Experimental Setup

The proposed SFNet can be used either in a speaker dependent or in a speaker independent setup. We evaluate it in a speaker independent manner in this work. The speech is pre-emphasized using $\kappa = 0.95$. Following this, we obtain the speech parameters comprising an 18 dimensional mel-cepstrum, pitch correlation [16] and pitch estimated using SWIPE algorithm in each frame [22]. We use one part of the TSP speech database for the training (≈4 hrs) having 20 speakers. We test on four (two male and two female) unseen speakers from the TSP speech database. We also test on the IIIT-H Indic Speech Database to examine the performance on the unseen language (Hindi and Kannada) speakers (twenty sentences for each language). We augment the original data by following the steps described in [16] and get 14 hrs of speech. We use LP order of $M = 25$ for the $A(z)$ filter. We use 40 filters of order 60 each for the noise filter bank, i.e., $Q = 40$ and $K = 60$. We use a FIR filter of order $P = 60$ for $C(z)$. In the experiments, Mel-spectrum of order 80 is used in the objective function. The proposed method is indicated by SFNet-U with U-units in the FeatureNet. We experiment with U={64,32}[2]. We experiment with two kinds of noise filter banks. In the first case, we fix the filter bank to the Gammatone filter bank [23]. In the second case, we learn the filter-bank as a part of the training and it is indicated by SFNet-U-L. The network weights and $\mathbf{c}$ are initialized randomly in case of SFNet-U. In the case of SFNet-U-L, the H is also initialized randomly. The network weights are trained using Adam optimizer [24]. Validation loss is monitored to stop the training process. The network is implemented using Keras and TensorFlow [25] with a batch size of 32.

The naturalness of the different methods is evaluated using the MUSHRA test [26] without the anchor similar to [16]. For the seen language test, we select 20 files randomly (5 from each of four test speakers) from each method and ask 10 listeners (5 males and 5 females) to rate in the range of 0-100 (0 being bad and 100 being excellent). For the unseen language test, we use three listeners from Kannada as well as Hindi to rate randomly selected 5 sentences in each language. As the load of listening increases with large number of test files in a listening test, a subjective listening test can evaluate only a small set of sentences. Hence, we use Perceptual Evaluation of Speech Quality (PESQ) [27] to quantitatively measure the quality of the synthesized speech for all test files[3]. To measure the complexity of the model, we compute the number of floating-point operations (FLOPS) required by the network that operates at the sampling rate. The number of FLOPS measures the time complexity and the number of parameters of the whole network (#params) measures the memory complexity of each algorithm.

[1] We experimentally found out the that l_2 loss does not perform well on voiced speech. There is lot more noise in the low-frequency region if l_2 loss is used.

[2] We did not find any improvement in naturalness for $U > 64$.

[3] The audio samples can be found in https://araomv.github.io/SFNet/

TABLE I
COMPARISON OF LPCNET AND DIFFERENT CONFIGURATIONS OF SFNET IN TERMS OF NUMBER OF PARAMETERS, PESQ AND QUALITY (MUSHRA) SCORE FOR SEEN (E) AND UNSEEN (U) LANGUAGES. THE VALUES IN THE BRACKET INDICATE THE STANDARD DEVIATION

model	#params	GFLOPs	PESQ (E)	PESQ (U)	Quality (E)	Quality (U)
Reference	-	-	-	-	98.19	100.0
LPCnet-192	602k	1.2	3.25(0.25)	2.49(0.46)	71.95(20.31)	44.30(22.47)
SFNet-64	50k	0.2	3.82(0.15)	**3.59 (0.10)**	79.81(17.37)	60.1(12.26)
SFNet-32	11k	0.2	3.80(0.14)	**3.56 (0.11)**	79.04 (18.22)	61.5(15.48)
SFNet-32-L	11k	0.2	**3.85(0.14)**	**3.59 (0.13)**	**82.04(16.95)**	61.6 (17.10)

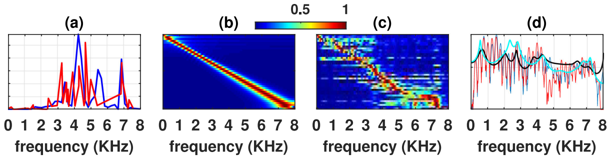

Fig. 2. (a) Median of filter gains (σ_r^2) predicted by the network at different center frequencies in the voiced region for SFNet-32 (blue) and SFNet-32-L (red) (b & c) Magnitude of frequency response of the 40 filters sorted by center frequency for SFNet-32 and SFNet-32-L. (d) Sample spectrum of a frame (blue) with spectrum of the predicted frame (red) along with the frequency response of the estimated LP filter using the SFNet (black) and Autocorrelation method (cyan)

C. Results

Table I shows a comparison of the proposed SFNet with different settings and the baseline LPCnet algorithm. It can be observed from the table that the PESQ using SFNet is better than that using LPCnet-192 for unseen speakers case. This indicates that the proposed excitation model is better than the general GRU based excitation model. There is slight improvement in PESQ when the U is increased from 32 to 64 (p <0.01). It can be observed from the table that the learned filter-bank is better than the fixed Gammatone filter bank ($p < 10^{-5}$) with $U = 32$. The mean value of PESQ reduces for the unseen language using all the methods compared to the case of English, but the percentage reduction in PESQ for the SFNet is ∼6%, while it is ∼23% for the LPCnet. There is no significant difference among the PESQ values on unseen language using three proposed models, namely SFNet-64, SFNet-32, SFNet-32-L. It is clear from the table that the quality score from the MUSHRA test is higher for seen language and the proposed method performs better than the LPCnet ($p < 10^{-6}$). It is also clear from the table that the quality score drastically reduces in the case of unseen language scenario, indicating a poor generalization of all methods across languages. The reduction in score from the MUSHRA test for the proposed SFNet is lower than that for the LPCnet.

The major complexity of the LPCnet method comes from the network that works at the sampling rate (arising due to a large number of multiplications between matrices of large sizes). In the proposed network the complexity arises only from the three filtering operations and interpolation. The complexity is given by $2 \times (P + 2 \times K \times Q + M) \times f_s$, where f_s is the sampling rate. For our experimental setup, the complexity turns out to be ≈160MFLOPs for the convolution operations and 40MFLOPs for other operations including excitation construction. This results in a total of 200MFLOPs (0.2GFLOPS). It can be observed that the number of parameters used by the SFNet, in comparison to that of the LPCnet, has reduced by 98%. This directly implies a reduction in memory footprint required by SFNet. This makes it possible to synthesize the real time audio on low-end devices with less memory and less power support.

D. Analysis of the Estimated Parameters

In this section we examine some of the parameters learned/predicted by the SFNet. Fig. 2(a) shows the median of the gain of the filters at each band across all voiced frames for the SFNet-32 (blue) and SFNet-32-L (red) case. It is clear from the figure that the energy at low frequencies (<3 kHz) is relatively lower than that at high frequencies. It is consistent with the signal processing based aperiodicity modeling algorithms such as WORLD [28] and STRAIGHT [15], where the band aperiodicity is modeled only above 3 kHz. It can be observed from the figure that a high aperiodicity occurs mostly around 4.2 kHz, 5.5 kHz, 7 kHz. The D4C method also models aperiodicity in the band of 3 kHz and 6 kHz to get natural speech [28]. Fig. 2(b) shows the magnitude response of the Gammatone filters used in SFNet-32. Fig. 2(c) shows the frequency response of the learned filters in SFNet-32-L sorted by their center frequencies. It is clear from the figure that most of the filters till 6 kHz are narrow bandpass filters which model the aperiodicity in the small bands even though there is no explicit constraint on the filters. The bandwidth of the filters above 6 kHz is wider compared to the filters below 6 kHz.

The LP coefficients are predicted from the network instead of estimating them from the mel-cepstrum or the autocorrelation method. Fig. 2(d) shows a sample and some of the predicted spectrum for a frame with a pitch value of 250Hz. It also shows the LP spectrum estimated using an auto-correlation as well as the LP spectrum computed from the predicted coefficients. It is clear from the figure that the LP spectrum predicted by the network is less biased by the nearest pitch harmonic unlike that from the autocorrelation based method [29]. This indicates that learning to predict the LP coefficients from the data enables to estimate a better envelope. The accuracy of LP spectrum is poor at high frequencies because of the Mel-spectrum based objective function for optimizing the network parameters.

V. CONCLUSION

In this paper, the source-filter model is reformulated where the source signal is further modeled as addition of two source signals. The first source signal is an impulse train constructed directly from the pitch sequence. The second source signal is modeled as colored noise using a filter bank with frequency dependent gains. We propose a neural network architecture called SFNet to predict the model parameters given a 20-dimensional speech feature vector. We show that the proposed method synthesizes speech having a quality better than that from the LPCnet using significantly less computational resources (memory and time). We also show that the parameters that are learned from the data are consistent with findings on source signals reported in the literature [30], [31]. As part of the future work, we plan to use the proposed method as a vocoder in a text-to-speech system. Future works also include adaption of the neural network with quantized features as inputs, evaluating the generalization of the SFNet in various conditions [32] as well as comparing with the conventional vocoders such as WORLD.

References

[1] Y. Wang *et al.*, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, 2017, pp. 4006–4010.

[2] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, no. 2, pp. 236–243, Apr. 1984.

[3] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: A vocoder-based high-quality speech synthesis system for real-time applications," *IEICE Trans. Inf. Syst.*, vol. 99, no. 7, pp. 1877–1884, 2016.

[4] J. Shen *et al.*, "Natural TTS synthesis by conditioning wavenet on Mel-spectrogram predictions," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 4779–4783.

[5] A. Gibiansky *et al.*, "Deep voice 2: Multi-speaker neural text-to-speech," in *Proc. Advances Neural Inf. Process. Syst.*, 2017, pp. 2962–2970.

[6] W. B. Kleijn *et al.*, "Wavenet based low rate speech coding," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 676–680.

[7] A. van den Oord *et al.*, "Wavenet: A generative model for raw audio," 2016, *arXiv:1609.03499*.

[8] A. van den Oord *et al.*, "Parallel WaveNet: Fast high-fidelity speech synthesis," in *Proc. 35th Int. Conf. Mach. Learn.*, 2018, pp. 3918–3926.

[9] N. Kalchbrenner *et al.*, "Efficient neural audio synthesis," in *Proc. 35th Int. Conf. Mach. Learn.*, 2018, pp. 2410–2419.

[10] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 3617–3621.

[11] L. Juvela, B. Bollepalli, V. Tsiaras, and P. Alku, "Glotnet—A raw waveform model for the glottal excitation in statistical parametric speech synthesis," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 6, pp. 1019–1030, Jun. 2019.

[12] L. Juvela, B. Bollepalli, J. Yamagishi, and P. Alku, "GELP: GAN-Excited linear prediction for speech synthesis from mel-spectrogram," in *Proc. Interspeech*, 2019, pp. 694–698.

[13] X. Wang, S. Takaki, and J. Yamagishi, "Neural source-filter-based waveform model for statistical parametric speech synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 5916–5920.

[14] J.-M. Valin and J. Skoglund, "A real-time wideband neural vocoder at 1.6kb/s using LPCNet," in *Proc. Interspeech*, 2019, pp. 3406–3410.

[15] H. 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.

[16] J. Valin and J. Skoglund, "LPCnet: Improving neural speech synthesis through linear prediction," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 5891–5895.

[17] G. Fant, *Acoustic Theory of Speech Production*. The Hague, The Netherlands: Mouton, 1960.

[18] S. M. Kay, *Modern Spectral Estimation: Theory and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[19] P. Kabal, "TSP speech database," *McGill University, Database Version: 2*, vol. 1, pp. 1–29, Nov. 2018.

[20] K. Prahallad, E. N. Kumar, V. Keri, S. Rajendran, and A. W. Black, "The IIIT-H Indic speech databases," in *Proc. 13th Annu. Conf. Int. Speech Commun. Assoc.*, 2012, pp. 2546–2549.

[21] J. Valin, "LPCNet implementation," 2020. Accessed: May 8, 2020. [Online]. Available: https://github.com/mozilla/LPCNet

[22] A. Camacho and J. G. Harris, "A sawtooth waveform inspired pitch estimator for speech and music," *J. Acoust. Soc. Amer.*, vol. 124, no. 3, pp. 1638–1652, 2008.

[23] M. Slaney, "An efficient implementaion of the Patterson-Holdsworth auditory filter bank," Apple Comput., Cupertino, CA, USA, Tech. Rep. 35, 1993.

[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[25] M. Abadi *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th {USENIX} Symp. Operating Syst. Des. Implementation*, 2016, pp. 265–283.

[26] Assembly, ITU Radiocommunication, "Methods for the subjective assessment of small impairments in audio systems including multichannel sound systems," Recommendation ITU-R BS, 1994.

[27] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 1, pp. 229–238, Jan. 2008.

[28] M. Morise, "D4C, a band-aperiodicity estimator for high-quality speech synthesis," *Speech Commun.*, vol. 84, pp. 57–65, 2016.

[29] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, no. 4, pp. 561–580, Apr. 1975.

[30] K. Tokuda and H. Zen, "Directly modeling speech waveforms by neural networks for statistical parametric speech synthesis," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 4215–4219.

[31] K. Tokuda and H. Zen, "Directly modeling voiced and unvoiced components in speech waveforms by neural networks," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 5640–5644.

[32] J. Lorenzo-Trueba *et al.*, "Towards achieving robust universal neural vocoding," in *Proc. Interspeech*, 2019, pp. 181–185.