

# Sequence-to-Sequence Acoustic Modeling for Voice Conversion

Jing-Xuan Zhang, Zhen-Hua Ling, *Member, IEEE*, Li-Juan Liu, Yuan-Jiang, and Li-Rong Dai

**Abstract**—In this paper, a neural network named Sequence-to-sequence ConvERsion NeTwork (SCENT) is presented for acoustic modeling in voice conversion. At training stage, a SCENT model is estimated by aligning the feature sequences of source and target speakers implicitly using attention mechanism. At conversion stage, acoustic features and durations of source utterances are converted simultaneously using the unified acoustic model, which is difficult to be achieved in conventional method. Mel-scale spectrograms are adopted as acoustic features which contain both excitation and vocal tract descriptions of speech signals. The bottleneck features extracted from source speech using an automatic speech recognition (ASR) model are appended as auxiliary input. A WaveNet vocoder conditioned on mel-spectrograms is built to reconstruct waveforms from the output of the SCENT model. Experimental results show that our proposed method achieved better objective and subjective performance than the baseline methods using Gaussian mixture models (GMM) and deep neural networks (DNN) as acoustic models. This proposed method also outperformed our previous work which achieved the top rank in Voice Conversion Challenge 2018. Ablation tests further confirm the effectiveness of appending bottleneck features and using attention module in our proposed method.

**Index Terms**—voice conversion, sequence-to-sequence, attention, mel-spectrogram.

## I. INTRODUCTION

VOICE conversion aims to modify the speech signal of a source speaker to make it sound like being uttered by a target speaker, while keeping the linguistic contents unchanged [1], [2]. The potential applications of this technique include entertainment, personalized text-to-speech, and so on [3], [4].

Building statistical acoustic models for feature mapping is a popular approach to voice conversion nowadays. At the training stage of the conventional voice conversion pipeline, acoustic features are firstly extracted from the waveforms of source and target utterances. Then, the features of parallel utterances are aligned frame by frame using alignment algorithms, such as dynamic time wrapping (DTW) [5]. Next, an acoustic model for conversion is trained using the acoustic features of paired source-target frames. The acoustic model can be a joint density Gaussian mixture model (JD-GMM) [3],

[6] or a deep neural network (DNN) [7], [8], both of which are universal function approximators [9], [10]. At the conversion stage, a mapping function is derived from the built acoustic model and converts the acoustic features of source speaker into those of target speaker. Finally, waveforms are recovered from the converted acoustic features using a vocoder.

This conventional pipeline for voice conversion has its limitations. First, most previous work focused on the conversion of spectral features and simply adjusted  $F_0$  trajectories linearly in the logarithm domain [7], [8], [11]–[15]. Besides, the durations of converted utterances were kept the same as the ones of source utterances since the acoustic models were built on a frame-by-frame basis. However, the production of human speech is a highly dynamic process and the frame-by-frame assumption constrains the modeling ability of mapping functions [16].

This paper proposes an acoustic modeling method for voice conversion based on the sequence-to-sequence neural network framework [17], [18]. A Sequence-to-sequence ConvERsion NeTwork (SCENT) is designed to directly describe the conditional probabilities of target acoustic feature sequences given source ones without explicit frame-to-frame alignment. The SCENT model follows the widely-used architecture of encoder-decoder with attention [19], [20]. The encoder network first transforms the input feature sequences into hidden representations which are suitable for the decoder to deal with. At each decoder time step, the attention module selects encoder outputs softly by attention probabilities and produces a context vector. Then, the decoder predicts output acoustic features frame by frame using context vectors. Furthermore, a post-filtering network is designed to enhance the accuracy of the converted acoustic features. Finally, a speaker-dependent WaveNet is utilized to recover time-domain waveforms from the predicted sequences of acoustic features.

In our proposed method, mel-scale spectrograms are adopted as acoustic features, which do not rely on the source-filter assumption of speech production. Therefore,  $F_0$  and spectral features are converted jointly in a single model. Additional bottleneck features derived using an automatic speech recognition (ASR) model are appended to the source mel-spectrograms, which are expected to improve the pronunciation correctness of the converted speech. Attention module learns the soft alignments between the pairs of source-target feature sequences implicitly. In conventional methods, the DTW algorithm is usually employed at data preprocessing stage in order to get frame-by-frame aligned training data. Compared to the deterministic DTW algorithm, seq2seq with attention is a statistical model which describes

This work was partially funded by National Key Research and Development Project of China (Grant No. 2017YFB1002202).

J.-X. Zhang, Z.-H. Ling and L.-R. Dai are with the National Engineering Laboratory for Speech and Language Information Processing, University of Science and Technology of China, Hefei, 230027, China (e-mail: nosisi@mail.ustc.edu.cn, zhling@ustc.edu.cn, lrdai@ustc.edu.cn). L.-J. Liu and Y. Jiang are with the iFLYTEK Co., Ltd., Hefei, 230088, China (e-mail: ljliu@iflytek.com, yuanjiang@iflytek.com).

This work was conducted when J.-X. Zhang was an intern at iFLYTEK Research.

the probabilities of output sequences given input ones at training time and is capable of predicting target acoustic sequences with durations different from source ones at conversion stage.

Experimental results show that our proposed method achieved better objective and subjective performance than the GMM-based and DNN-based baseline systems. This proposed method also outperformed our previous work which achieved the top rank in Voice Conversion Challenge 2018 [21]. Ablation tests further confirm the effectiveness of appending bottleneck features and using the attention module.

The rest of this article is organized as follows. Section II reviews the related work on seq2seq modeling and WaveNet vocoders. Section III introduces our proposed method for voice conversion. Details and results of experiments are presented in Section IV. The article is concluded in Section V.

## II. RELATED WORK

### A. Relationship with sequence-to-sequence learning for text-to-speech

Text-to-speech (TTS) methods based on seq2seq learning have emerged recently and attracted much attention [22]–[25]. Our work is inspired by the success of applying seq2seq models to TTS. However, voice conversion is different from TTS in several aspects. First, the inputs of a voice conversion model are frame-level acoustic features rather than phone-level or character-level linguistic features. Typically, linguistic features are discrete, while acoustic features are continuous. In addition to linguistic information, acoustic features also contain speaker identity information which should be processed during voice conversion. Second, the input-output alignment in voice conversion task is different from that in TTS. Speech generation in TTS is a *decompressing* process and the alignment between text and acoustic frames is usually a *one-to-many* mapping. While the alignment can be either *one-to-many* or *many-to-one* in voice conversion, depending on the characteristics of speaker pairs and the dynamic characteristics of acoustic sequences. Third, the training data available for voice conversion is typically smaller than that for TTS.

### B. Sequence-to-sequence learning for voice conversion

To the best of our knowledge, Ramos [26] made the first attempt to convert spectral features using a sequence-to-sequence model with attention. Although the model worked well given ground truth target spectral features as history contexts, it failed to generate real-valued output predictions using predicted history contexts. Kaneko *et al.* [27] proposed a CNN-based seq2seq spectral conversion method. Because of the lack of attention module in their method, the DTW algorithm was still utilized in order to obtain frame-level aligned feature sequences during training data preparation. Miyoshi *et al.* [28] proposed a method of mapping context posterior probabilities using seq2seq models. In their method, an RNN-based encoder-decoder converted the source posterior probability sequence to the target one for each phone, and the phone durations of natural target speech were necessary at conversion stage.

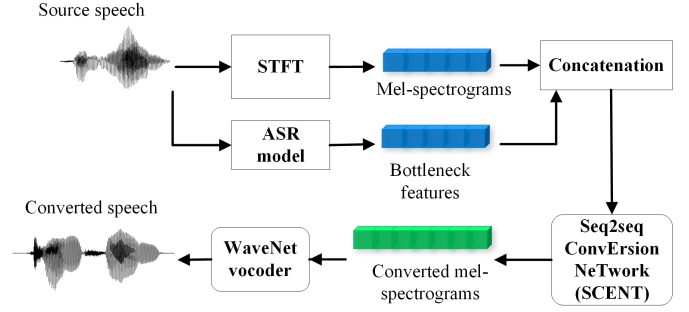


Fig. 1. The conversion process of our proposed sequence-to-sequence voice conversion method.

Our work is most similar to Ramos’s one [26], where an utterance-level seq2seq with attention model is built for acoustic feature conversion. Different from previous methods, mel-spectrograms are adopted as acoustic features in our method, thus  $F_0$  and spectral features are transformed jointly. Our method has the ability of modeling pairs of input and output utterances without dependency on DTW alignment. During conversion, the durations of generated target acoustic sequences are determined automatically and the probability of completion is predicted at each decoder time step.

### C. Voice conversion using WaveNet

WaveNet [29], as a neural network-based waveform generation model, has been successful applied to TTS and voice conversion areas [21], [30], [31]. Studies have shown that WaveNet vocoders outperformed traditional vocoders such as World [32] and STRAIGHT [33] in terms of the quality of reconstructed speech [21], [34], [35]. Voice conversion methods using WaveNet models have also been studied in recent years. Kobayashi *et al.* [30] proposed a GMM-based voice conversion method with WaveNet-based waveform generation. Liu *et al.* [21] proposed building WaveNet vocoders for voice conversion with limited data by model adaptation. Directly mapping source acoustic features into target speaker’s waveforms using WaveNet has also been proposed [31].

In this paper, WaveNet vocoders are used to reconstruct the waveforms of target speakers. WaveNet vocoders accept mel-spectrograms as input conditions and are trained in a speaker-dependent way without using the adaptation technique described in [21].

## III. PROPOSED METHOD

### A. Overall architecture

Fig. 1 shows the diagram of our proposed method when converting an input utterance. The conversion process can be divided into two main stages. One is a Seq2seq ConvErsion NeTwork (SCENT) for acoustic feature prediction, the other is a WaveNet neural vocoder for waveform generation. Mel-spectrograms are adopted as acoustic features in this paper. Bottleneck features exacted by an ASR model from source speech are concatenated with acoustic features to form the input sequences of the SCENT model. The SCENT model converts input sequence into mel-spectrograms of the target

speaker. Then, the target speaker's speech is synthesized by passing the predicted mel-spectrograms through the WaveNet vocoder.

### B. Feature extraction

Mel-spectrograms are computed through a short-time Fourier transform (STFT) on waveforms. The STFT magnitudes are transformed to mel-frequency scale using mel-filterbanks followed by a logarithmic dynamic range compression. In order to extract bottleneck features, a recurrent neural network (RNN) based ASR model is trained on a separate speech recognition dataset. For each input frame, bottleneck features, i.e., the activations of the last hidden layer before the softmax output layer of the ASR model, are extracted. Such bottleneck features can provide additional linguistic related descriptions which are expected to benefit the conversion process. It should be noticed that these bottleneck features are still automatically extracted from the acoustic signals of source utterances and no text transcriptions are necessary. The mel-spectrograms and bottleneck features at each frame are concatenated to form the input sequence  $\mathbf{x} = [\mathbf{x}_1, \dots, \mathbf{x}_{T_x}]$  of the SCENT model, where  $T_x$  is the frame number of source speech.

### C. Structure of SCENT

A SCENT model contains an encoder-decoder with attention network which predicts acoustic feature in an uni-directional left-to-right way, and a bi-directional post-filtering network which refines the generation results. Fig. 2 shows the network structure of a SCENT model.

Let  $\mathbf{y} = [\mathbf{y}_1, \dots, \mathbf{y}_{T_y}]$  denote the output mel-spectrogram sequence of the encoder-decoder network, where  $T_y$  is the frame number of target speech. The encoder-decoder network models the mapping relationship between input and output feature sequences using conditional distributions of each output frame  $\mathbf{y}_t$  given previous output frames  $\mathbf{y}_{<t} = [\mathbf{y}_1, \dots, \mathbf{y}_{t-1}]$  and the input  $\mathbf{x}$  as

$$p(\mathbf{y}|\mathbf{x}) = \prod_{t=1}^{T_y} p(\mathbf{y}_t|\mathbf{y}_{<t}, \mathbf{x}, W_{enc}, W_{dec}), \quad (1)$$

where  $W_{enc}$  and  $W_{dec}$  are parameters of the encoder-decoder network. As shown in Fig. 2, the encoder transforms the concatenated mel-spectrograms and bottleneck features of source speech into a high-level and abstract representation  $\mathbf{h} = [\mathbf{h}_1, \dots, \mathbf{h}_{T_h}]$  as

$$\mathbf{h} = \text{Encoder}(\mathbf{x}, W_{enc}). \quad (2)$$

The decoder with attention mechanism utilizes  $\mathbf{h}$  and produces a probability distribution over output frames as

$$p(\mathbf{y}_t|\mathbf{y}_{<t}, \mathbf{x}) = \text{Decoder}(\mathbf{y}_{<t}, \mathbf{h}, W_{dec}). \quad (3)$$

The generation process of the decoder network is uni-directional. In order to make use the bi-directional context information, a post-filtering network (i.e., PostNet) is further employed to enhance the accuracy of prediction. Let  $\mathbf{z} = [\mathbf{z}_1, \dots, \mathbf{z}_{T_z}]$  represents the PostNet output sequence, which

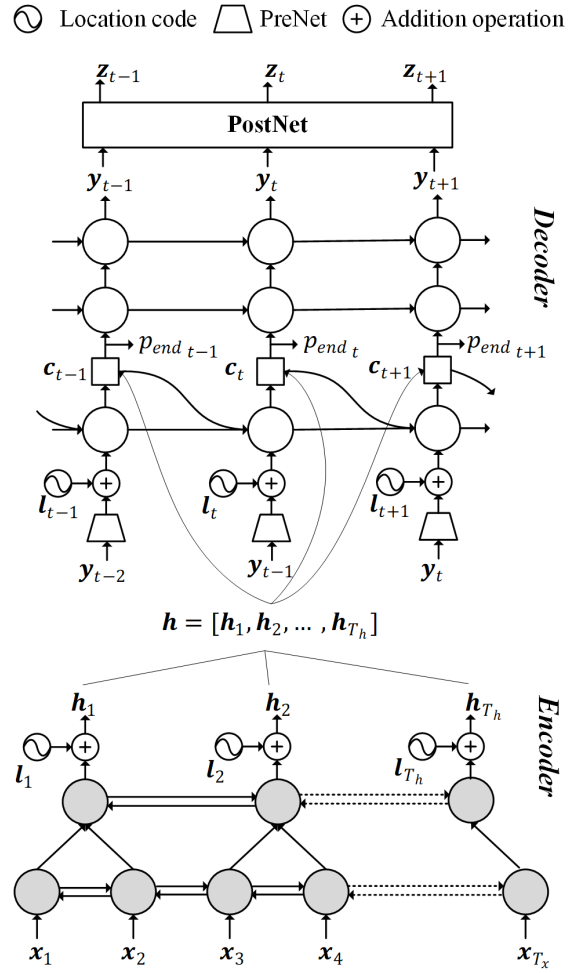


Fig. 2. The network structure of a SCENT model, where skip connections and residual connections are ignored for clarity. The grey circles in the encoder represent LSTM units with layer normalization. The encoder in this figure has a downsampling rate  $M = 2$ . The auto-regressive inputs of the decoder are natural history contexts at training time and are generated ones at conversion time. Single frame is predicted at each decoder time step (i.e.,  $r = 1$ ) in this figure.

is the final prediction of the SCENT model. Typically, we have  $T_z = T_y$ . The distribution of feature sequence  $\mathbf{z}$  given the output of the encoder-decoder network  $\mathbf{y}$  is modeled as

$$p(\mathbf{z}|\mathbf{y}) = \text{PostNet}(\mathbf{y}, W_{pos}), \quad (4)$$

where  $W_{pos}$  denotes the parameters of the PostNet. Next, we will describe each part of SCENT in details.

1) *Encoder*: The encoder network is constructed based on the pyramid bidirectional LSTM architecture [36], [37], which processes the sequence with lower time resolution at higher layers. In a conventional deep bidirectional LSTM (BLSTM) architecture, the output at the  $n$ -th time step of the  $j$ -th layer is computed as

$$\mathbf{h}_n^j = \text{BLSTM}(\mathbf{h}_{n-1}^j, \mathbf{h}_n^{j-1}). \quad (5)$$

In a pyramid BLSTM (pBLSTM), the outputs at consecutive steps of a lower layer are concatenated and fed into the next layer to decrease the sampling rate of input sequence.

The general calculation of pBLSTM hidden units can be formulated as

$$\mathbf{h}_n^j = \text{pBLSTM}(\mathbf{h}_{n-1}^j, [\mathbf{h}_{M*n}^{j-1}, \dots, \mathbf{h}_{M*n+M-1}^{j-1}]), \quad (6)$$

where  $M$  is ratio of downsampling. The technique of layer normalization [38] is applied to the encoder LSTM cells. Then, a location code  $\mathbf{l}_n = [l_n(0), \dots, l_n(d-1)]^\top$  [39] is added to the top output layer of pBLSTMs to form the hidden representation  $\mathbf{h}$ . Let  $d$  be the dimension of each  $\mathbf{h}_n$ . The location code is composed of sine and cosine functions of different frequencies as

$$\begin{cases} l_n(2i) = \sin(n/10000^{2i/d}), \\ l_n(2i+1) = \cos(n/10000^{2i/d}), \end{cases} \quad (7)$$

where  $n$  is the time step in sequence  $\mathbf{h}$  and  $i \in [0, \dots, d/2-1]$  is the dimension index. This location code is useful since it gives the model explicit information of which portion of the sequence is currently processed.

The pyramid structure of our encoder network results in shorter hidden representation than original input sequence. For the voice conversion task, we expect that the encoder network should exclude speaker-dependent information of the source speech and extract hidden representation  $\mathbf{h}$  which is high-level and linguistic-related. Because one phone usually corresponds to tens of acoustic frames, it is reasonable to derive hidden representation with lower sampling rate than the frame-level input sequence. Furthermore, hidden representation with lower sampling rate makes the attention module easier to converge, since this leads to fewer encoding states for attention calculation at each decoding step. This pyramid structure also reduces the computational complexity by shortening the length of  $\mathbf{h}$  for attention calculation and speeds up training and inference significantly.

2) *Decoder with attention mechanism*: The decoder is an auto-regressive RNN which predicts the output acoustic features from the hidden representation  $\mathbf{h}$ . Non-overlapping  $r$  frames are predicted at each decoder step. This trick divides the total decoding steps by  $r$ , which further reduces training and inference time [22]. In Fig. 2, the decoder is illustrated with  $r = 1$  for clarity. The prediction of previous time step  $\mathbf{y}_{t-1}$  is firstly passed through a pre-processing network (i.e., PreNet), which is a two-layer MLP with ReLU activation and dropout in our implementation. The MLP outputs are sent into an LSTM layer with attention mechanism. A context vector  $\mathbf{c}$  is calculated at each decoder step using attention probabilities as

$$\mathbf{c}_t = \sum_{n=1}^{T_h} \alpha_t^n \mathbf{h}_n, \quad (8)$$

where  $\alpha_t = [\alpha_t^1, \dots, \alpha_t^{T_h}]$  are the attention probabilities,  $t$  is decoder time step, and  $n$  is the index of encoder outputs.

In our implementation, a hybrid attention mechanism is adopted which takes the alignment of previous decoder step (i.e., location-awareness) into account when computing the attention probabilities. In order to exact location information,  $k$  filters with kernel size  $l$  are employed to convolve the alignment of previous time step. Let  $F \in \mathbb{R}^{k \times l}$  represent the convolution matrix, and  $\mathbf{q}$  denote the query vector, which is

given by output of attention LSTM. Then, the attention score  $e_t^n$  is computed as

$$\mathbf{f}_t = F * \alpha_{t-1}, \quad (9)$$

$$e_t^n = \mathbf{q}_t^\top W \mathbf{h}_n + \mathbf{v}^\top \tanh(U \mathbf{f}_t^n + \mathbf{b}). \quad (10)$$

Furthermore, the forward attention method proposed in our previous work [40] is adopted to stabilize the attention alignment and speed up the convergence of attention alignment. In the forward attention method, the attention probability  $\alpha_t^n$  is calculated as

$$\hat{e}_t^n = \exp(e_t^n) / \sum_{i=1}^{T_h} \exp(e_t^i), \quad (11)$$

$$\hat{\alpha}_t^n = \hat{e}_t^n (\alpha_{t-1}^n + \alpha_{t-1}^{n-1}), \quad (12)$$

$$\alpha_t^n = \hat{\alpha}_t^n / \sum_{i=1}^{T_h} \hat{\alpha}_t^i. \quad (13)$$

For initialization, we have

$$\begin{cases} \alpha_0^1 = 1, \\ \alpha_0^n = 0, \end{cases} \text{ for } n = 2, \dots, T_h. \quad (14)$$

In addition, a location code is also added to the auto-regressive input of the decoder at each time step.

The context vector  $\mathbf{c}$  and query vector  $\mathbf{q}$  are concatenated and fed into a stack of two-layer decoding LSTMs. The concatenation of  $\mathbf{c}$ ,  $\mathbf{q}$  and outputs of decoding LSTMs are linearly projected to produce the mel-spectrogram output of the decoder network. In parallel, the concatenation of  $\mathbf{c}$  and  $\mathbf{q}$  are linearly projected to a scalar and passed through a sigmoid activation to predict the completion probability  $p_{end}$ , which indicates whether the converted sequence reaches the last frame.

3) *Post-filtering network*: The PostNet refines the mel-spectrograms predicted by the decoder using bi-directional context information. The PostNet is a convolutional neural network (CNN) with a residual connection from network input to the final output. The first layer of the PostNet is composed of 1-D convolution filter banks in order to extract rich context information. The outputs of the convolutional banks are stacked together and further passed through a two-layer 1-D CNN. The outputs of the final layer are added to the input mel-spectrograms to produce the final results.

#### D. Loss function of SCENT

We train the SCENT model by multi-task learning and the total loss is the weighted sum of three sub-losses as

$$L = w_{dec} L_{dec} + w_{post} L_{post} + w_{end} L_{end}, \quad (15)$$

where  $w_{dec}$ ,  $w_{post}$  and  $w_{end}$  are the weights of the three components.  $L_{dec}$  and  $L_{post}$  denote the losses of mel-spectrogram prediction given by the decoder and the PostNet respectively.  $L_{end}$  is a binary cross-entropy loss for evaluating the predicted completion probabilities.

Two types of criteria are investigated for  $L_{dec}$ . One is the minimum square error (MSE) between the predicted and ground truth acoustic features. The other is the maximum

likelihood (ML) criterion based on Gaussian mixture model (GMM). For GMM-ML, the network outputs are adopted to parameterize a GMM following the framework of mixture density networks (MDN) [41]. Under both criteria, natural acoustic histories of target speech are sent into the decoder at training time. The MSE criterion is actually a special case of GMM-ML which uses single mixture with fixed unit variance and predicted mean vector [42]. Theoretically, GMM-ML is more flexible since it models more general probability distributions and the MSE criterion usually leads to over-smoothed prediction because the averaging effect [41].

When applying the GMM-ML criterion to  $L_{dec}$ , the mean vector of the component with maximum prior probability is used to generate the output sample at both training and testing stages. At training time, the gradients from the PostNet are only back-propagated through the sampled mean vectors given by the decoder output layer.

Only the MSE criterion is applied to  $L_{post}$  in our implementation. For calculating  $L_{end}$ , only the last decoder step of a natural target sequence is labelled as 1 (i.e., completed) and the rest steps are labelled as 0 (i.e., incompleting).

#### E. WaveNet-based vocoder

As shown in Fig. 1, a WaveNet-based vocoder is adopted to reconstruct time-domain waveforms given the predicted mel-spectrogram features. In our WaveNet model, the mel-spectrogram features are firstly passed through a ConditionNet consisting of stack of dilated 1-D convolution layers with parametric ReLU activation (PReLU) [43]. The outputs of ConditionNet are upsampled to be consistent with the sampling rate of waveforms by simply repeating. Then, the sequence of condition vectors are fed into each dilated convolution block of the WaveNet to control the waveform generation. Our WaveNet model is trained only using the target speech data for building the SCENT model and the adaptation technique [21] is not used in this paper.

### IV. EXPERIMENTS

#### A. Experimental conditions

Our dataset for experiments contained 1060 parallel Mandarin utterances from one male speaker (about 53 min) and one female speaker (about 72min). This dataset was separated into a training set with 1000 utterances, a validation set with 30 utterances and a test set with 30 utterances. The recordings were sampled at 16kHz. 80-dimensional mel-scale spectrograms were extracted every 10 ms with Hann windowing of 50 ms frame length and 1024-point Fourier transform. 512-dimensional bottleneck features were extracted using an ASR model every 40 ms and were then upsampled by repeating to match the frame rate of mel-spectrograms.

The speaker-independent ASR model was trained using an internal dataset of iFLYTEK company, which contained about 10,000 hours of recordings with aligned phonetic transcriptions. The target of the ASR model is clustered tri-phonemes senones. The phonemes set which constitute the target of the ASR model include 26 initials and 35 kind of vowels. Each vowel is with 4 kind of tones, thus the phonemes set contains

TABLE I  
DETAILS OF MODEL CONFIGURATIONS.

SCENT	Encoder	pBLSTM, 2 layers and 256 cells LSTM with layer normalization, $M = 4$
	PreNet	FC-256-ReLU-Dropout(0.5)→FC-256-ReLU-Dropout(0.5)
	Decoder	Attention LSTM, 1 layer and 256 cells; $k = 10$ and $l = 32$ for $F$ in Eq. (9); $v$ in Eq. (10) has dimension of 256; Decoder LSTM, 2 layers and 256 cells
	PostNet	Conv1D banks, $k = [1, \dots, 8]$ , Conv1D- $k$ -256-BN-ReLU-Dropout(0.2)→Conv1D-3-256-BN-ReLU-Dropout(0.2)→Conv1D-3-256-BN-ReLU-Dropout(0.2)
WaveNet vocoder	ConditionNet	4 layers Conv1D-3-100-PReLU with dilation $d = [1, 2, 4, 8]$
	WaveNet	30 layers dilated convolution layers with dilation $d = 2^{k \bmod 10}$ for $k = [0, \dots, 29]$ ; 1024 softmax output

FC represents fully connected. BN represents for batch normalization. Conv1D- $k$ - $n$  represents 1-D convolution with kernel size  $k$  and channel  $n$ .

totally 166 categories. We evaluated the performance of the ASR model on the parallel dataset for voice conversion. The frame accuracies for the female and male speaker were 72.3% and 78.4% respectively.

The details of our model configurations are listed in TABLE I. In our implementation, two frames were predicted at one decoding step (i.e.,  $r = 2$ ) and only the last frame was fed back into the PreNet for the generation at next step. In the loss function for training the SCENT model,  $w_{dec}$  was heuristically set as 1.0 or 0.01 if MSE or GMM-ML training criterion was adopted for  $L_{dec}$ .  $w_{post}$  and  $w_{end}$  were heuristically set as 1.0 and 0.005 respectively. Zoneout [44] with probability of 0.2 were used at LSTM layers for regularization. Residual connections were adopted for the LSTM layers of encoder and decoder to speed up model convergence. We used Adam [45] optimizer with learning rate of  $10^{-3}$  for the first 20 epochs. After 50 epochs, the learning rate was exponentially decayed by 0.95 for each epoch.  $L_2$  regularization with weight  $10^{-6}$  was also applied. The batch size was 4. For WaveNet training, the  $\mu$ -law companded waveforms were quantized into 10 bits, i.e., 1024 levels. A speaker-dependent WaveNet vocoder was trained using each speaker's waveforms with random initialization and a learning rate of  $10^{-4}$  until the loss converge.

Three kind of baseline methods were adopted for comparison in our experiments. 41-dimensional mel-cepstral coefficients (MCCs), 1-dimensional fundamental frequency ( $F_0$ ) and 5-dimensional band aperiodicities (BAPs) were extracted every 5 ms by STRAIGHT [33] as acoustic features in our baseline systems. The descriptions of these methods are as follows<sup>1</sup>.

- **JD-GMM:** Gaussian mixture models with full-covariance matrices were utilized for modeling the joint spectral feature vectors of source and target speakers. For each speaker, static and delta spectral features were used.

<sup>1</sup>Samples of audio are available at <https://jxzhanggg.github.io/Seq2SeqVC>.

TABLE II  
OBJECTIVE EVALUATION RESULTS OF USING DIFFERENT LOSS FUNCTIONS  
FOR THE DECODER ON VALIDATION SET.

Setting	Female-Male		Male-Female	
	MCD (dB)	$F_0$ RMSE (Hz)	MCD (dB)	$F_0$ RMSE (Hz)
MSE	3.397	42.122	3.658	33.420
MX2	<b>3.365</b>	<b>38.123</b>	<b>3.649</b>	<b>32.271</b>
MX4	3.384	38.629	3.651	34.748
MX6	3.376	38.804	3.669	35.337
MX8	3.418	39.230	3.637	33.029

“MX2”, “MX4”, “MX6” and “MX8” represent using ML criterion with 2, 4, 6 and 8 GMM mixture components respectively.

The number of mixtures  $m$  was tuned on validation set with  $m \in [16, 32, 48, 64]$ . Maximum likelihood parameters generation (MLPG) with global variance (GV) enhancement were used for spectral parameter generation.  $F_0$  was converted by Gaussian normalization in the logarithm domain [46]. BAPs were not converted but directly copied from the source, since previous research showed that converting aperiodic component did not make statistically significant difference on the quality of converted speech [47]. Waveforms were reconstructed by STRAIGHT vocoder from the converted acoustic features.

- **DNN:** The DNN-based voice conversion models were implemented based on Merlin toolkit [48]. The static, delta and acceleration components of MCCs,  $F_0$  and BAPs features were transformed jointly using a DNN. In addition to use the acoustic features of the source speaker as model input, we also concatenated the input acoustic features with the bottleneck features used in our proposed method. This approach was named **bn-DNN** in the rest of this paper. The ReLU activation function was used at DNN hidden units. A grid search using validation set was adopted in order to pick up the optimal depth  $d$  and width  $w$  of the DNN with  $d \in [3, 4, 5, 6]$  and  $w \in [512, 1024, 2048]$ . MLPG and GV techniques were used for acoustic parameter generation. Waveform was reconstructed by STRAIGHT vocoder from the converted acoustic features.
- **VCC2018:** This baseline method followed the framework of our previous work [21], which achieved the top rank in Voice Conversion Challenge 2018. A speaker-dependent acoustic feature predictor was trained using the data of the target speaker. The predictor was an LSTM model which predicted MCCs,  $F_0$  and BAPs features of the target speaker from its bottleneck features frame-by-frame. In this method, a speaker-dependent WaveNet vocoder conditioned on MCCs,  $F_0$  and BAPs features was built for waveform reconstruction.

#### B. Comparison between different decoder loss functions

As introduced in Section III-D, either MSE or GMM-ML criterion was applied to define the loss function  $L_{dec}$  of the decoder output in our implementation. We evaluated the

objective performance of these loss functions by experiments on both female-to-male and male-to-female conversions. The mel-cepstral distortion (MCD) and root mean square error (RMSE) of  $F_0$  on validation set were adopted as metrics. Because mel-spectrograms were adopted as acoustic features, it's not straightforward to extract  $F_0$  and MCCs features from the converted acoustic features. Therefore,  $F_0$  and 25-dimensional MCCs features were extracted by STRAIGHT from the reconstructed waveforms for evaluation. Then the extracted features were aligned to those of the reference utterances in the validation set in order to compute MCD and  $F_0$  RMSE values. The  $F_0$  RMSE was calculated only using the frames which were both voiced in the converted and reference utterances.

TABLE II summarizes the objective evaluation results on validation set. From the table, we can see that the model using the GMM-ML criterion with 2 mixture components achieved the best performance on validation set among all settings. A further examination shows that using the GMM-ML criterion with mixture components more than 2 may lead to the instability of attention alignment. Some cases of attention failures, such as getting stuck in one frame, can be observed for MX6 and MX8. One possible reason is that larger mixture numbers may increase the number of parameters and the difficulty of model training. Thus, the GMM-ML criterion with 2 mixtures was adopted for  $L_{dec}$  in following experiments.

An example of the alignment between an utterance pair using the SCENT model is shown in Fig. 3, where each column denotes the attention probabilities corresponding to different encoder states for one decoder step. The DTW algorithm was also conducted based on the input and output mel-spectrogram sequences and the resulting path was plotted as the red dashed line for comparison. From this figure, we can see that these two alignments matched well. Comparing with the DTW path which denotes hard and deterministic alignment, the attention alignment is soft and changes smoothly along consecutive decoder time steps.

#### C. Comparison between baseline and proposed methods

1) *Objective evaluation:* Objective evaluations were first carried out to compare the MCD and  $F_0$  RMSE performance of our proposed method and the baseline methods introduced in Section IV-A, including JD-GMM, DNN, bn-DNN and VCC2018. In order to compensate the duration differences between source and target speakers, we also tried to linearly interpolate the source feature sequences before sending them into the conversion models according to the average ratio between the training set durations of the two speakers. This interpolation was conducted only to the static source features and the dynamic features were recalculated based on the interpolated static features. This led to four additional methods, named i-JD-GMM, i-DNN, i-bn-DNN and i-VCC2018, in our evaluation. The MCDs and  $F_0$  RMSEs were calculated following the way introduced in Section IV-B. For fair comparison,  $F_0$  and MCCs features were re-extracted by STRAIGHT from the converted waveforms for all methods when computing MCDs and  $F_0$  RMSEs.



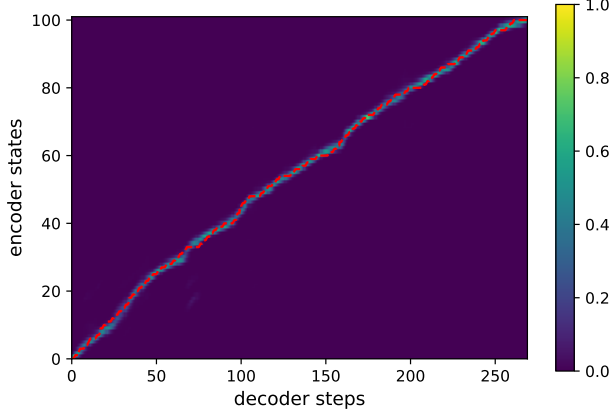


Fig. 3. Visualization of the attention alignment and the DTW path of an utterance pair in the validation set. The heat map shows the alignment probabilities calculated by the attention module in our seq2seq model. The red dashed line shows the alignment path given by DTW, which is downsampled to match the sample rates of encoder states and decoder time steps.

TABLE III  
OBJECTIVE EVALUATION RESULTS OF BASELINE AND PROPOSED METHODS ON TEST SET.

Methods	Female-Male		Male-Female	
	MCD (dB)	$F_0$ RMSE (Hz)	MCD (dB)	$F_0$ RMSE (Hz)
JD-GMM	3.892	55.241	4.307	46.625
i-JD-GMM	3.936	55.939	4.328	48.286
DNN	3.688	44.087	4.335	39.190
i-DNN	3.750	44.268	4.245	39.877
bn-DNN	3.618	42.385	4.078	35.883
i-bn-DNN	3.725	42.961	4.088	35.019
VCC2018	3.802	56.874	4.210	39.196
i-VCC2018	3.854	53.350	4.225	41.257
Proposed	<b>3.556</b>	<b>41.748</b>	<b>3.802</b>	<b>33.374</b>

“i” represents the interpolation of source features for duration compensation. “bn” denotes appending bottleneck features as input.

TABLE III shows the objective evaluation results of baseline and proposed methods on test set. We can see that the MCDs and  $F_0$  RMSEs of baseline methods with interpolation were closed to or slightly worse than those without interpolation. Appending bottleneck features as input was beneficial for improving the objective performance of the DNN-based method. Our proposed method outperformed all baseline methods, which obtained the lowest MCD and  $F_0$  RMSE.

One advantage of our proposed method is that it can convert the duration of source speech using an unified acoustic model. In order to investigate the performance of duration conversion, the scatter diagrams of test utterance durations are drawn in Fig. 4 and Fig. 5 for female-to-male and male-to-female conversion respectively. For each test utterance, the durations of speech converted using different baseline methods were the same, i.e., the duration of the source speech. For the baseline methods with source feature interpolation, the same global interpolation ratio was shared by all baseline methods. Therefore, “i-Baseline” and “Baseline” in these two figures stand for all baseline methods with and without

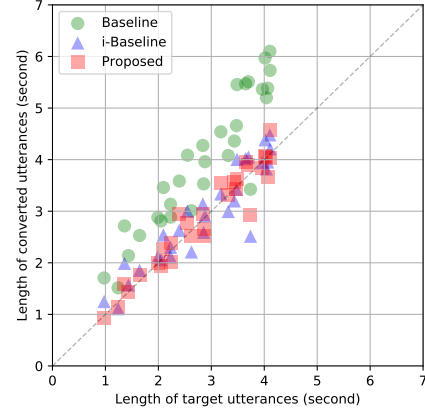


Fig. 4. The scatter diagram of the durations of test utterances for female-to-male conversion.

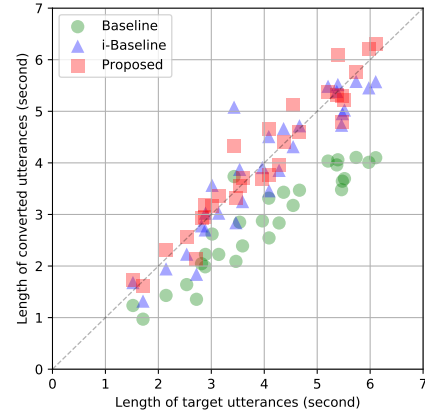


Fig. 5. The scatter diagram of the durations of test utterances for male-to-female conversion.

TABLE IV  
THE AVERAGE ABSOLUTE DIFFERENCES BETWEEN THE DURATIONS OF THE CONVERTED AND TARGET UTTERANCES ON TEST SET.

Methods	Female-Male (second)	Male-Female (second)
Baseline	1.147	1.157
i-Baseline	0.276	0.380
Proposed	<b>0.194</b>	<b>0.260</b>

interpolation respectively. From these figures, we can see that the male speaker had higher speaking rate and shorter utterance durations than the female speaker. The simple linear interpolation made the length of converted speech closer to the target. Furthermore, the average absolute differences between the durations of the converted and target utterances were calculated and presented in TABLE IV. Results show that our proposed method can generate speech with lower duration errors than the baseline methods without duration modification or with global speaking rate compensation.

Fig. 6 plots the  $F_0$  contours and spectrograms of one test utterance converted using different methods and the natural target reference. From this figure, we can see that our proposed method can generate speech with more similar  $F_0$

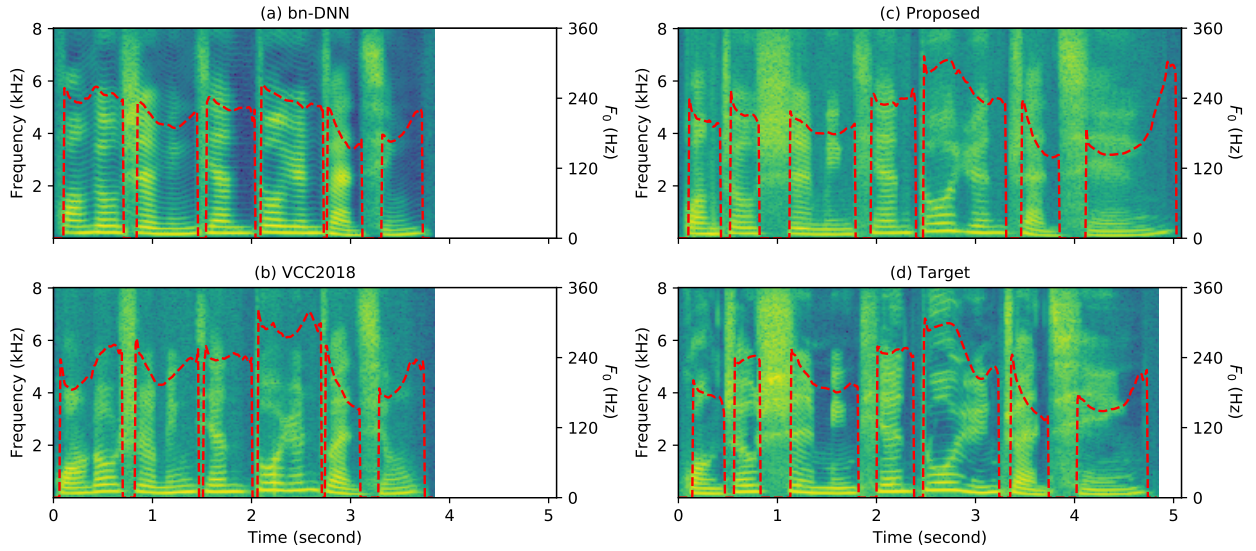


Fig. 6. The  $F_0$  contours and spectrograms of one test utterance converted using different methods and the natural target reference. The red dashed lines are  $F_0$  contours extracted by STRAIGHT from the converted waveforms.

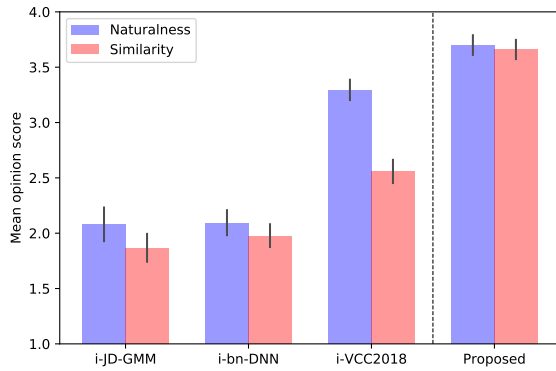


Fig. 7. Mean opinion scores (MOS) on naturalness and similarity of baseline and proposed methods for female-to-male conversion. Error bars indicate 95% confidence intervals.

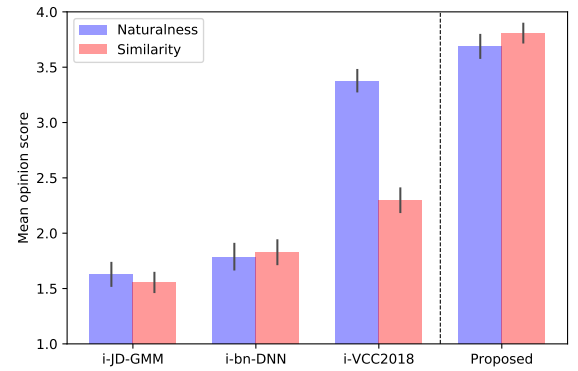


Fig. 8. Mean opinion scores (MOS) on naturalness and similarity of baseline and proposed methods for male-to-female conversion. Error bars indicate 95% confidence intervals.

contours to the natural reference than the other two baseline methods. Furthermore, our proposed method can also modify the duration of source speech towards the natural reference appropriately as shown in this figure.

2) *Subjective evaluation*: Subjective evaluations were conducted to compare performance of our proposed method with the baseline methods in terms of both naturalness of similarity of converted speech. In this evaluation, twenty utterances in the test set were randomly selected and converted using our proposed method and three baseline methods, including i-JD-GMM, i-bn-DNN, and i-VCC2018. Ten native listeners participated in the evaluation and the samples were shown to them in random order. The listeners were asked to give a 5-scale opinion score (5: excellent, 4: good, 3: fair, 2: poor, 1: bad) on both similarity and naturalness for each converted utterance.

The results of the subjective evaluations for female-to-male and male-to-female conversions are presented in Fig. 7 and

Fig. 8 respectively. From these two figures, we can see that the i-bn-DNN method achieved similar naturalness to and better similarity than the i-JD-GMM method. This is consistent with previous studies on DNN-based voice conversion methods [7], [8], [11]. It should be noticed that the i-bn-DNN method accepted additional bottleneck features as inputs, which may benefit the performance of this method. Compared with the i-bn-DNN method, the i-VCC2018 method did not use acoustic features as input. However, this method achieved the best performance among the three baseline methods, especially on the naturalness of converted speech. One important reason is that the i-VCC2018 method adopted WaveNet vocoder instead of conventional STRAIGHT vocoder to reconstruct speech waveforms from the converted acoustic features.

Our proposed method outperformed the i-VCC2018 method on both naturalness and similarity in both figures. One possible reason is that at the conversion stage of the i-VCC2018 method, bottleneck features extracted from source speech



TABLE V  
OBJECTIVE EVALUATION RESULTS OF PROPOSED METHODS WITH AND WITHOUT USING BOTTLENECK FEATURES.

Methods	Female-Male		Male-Female	
	MCD (dB)	$F_0$ RMSE (Hz)	MCD (dB)	$F_0$ RMSE (Hz)
w-bn	<b>3.556</b>	<b>41.748</b>	<b>3.802</b>	<b>33.374</b>
w/o-bn	3.624	48.550	4.000	40.183

“w-bn” and “w/o-bn” represent the models with and without using bottleneck features respectively.

TABLE VI  
OBJECTIVE EVALUATION RESULTS OF PROPOSED METHODS WITH AND WITHOUT THE ATTENTION MODULE.

Methods	Female-Male		Male-Female	
	MCD (dB)	$F_0$ RMSE (Hz)	MCD (dB)	$F_0$ RMSE (Hz)
w-att	<b>3.556</b>	<b>41.748</b>	<b>3.802</b>	<b>33.374</b>
w/o-att	3.635	47.620	3.969	37.948
i-w/o-att	3.770	50.310	3.914	37.034

“w-att”, “w/o-att” and “i-w/o-att” represent models with attention module, without attention module and without attention module but adjusting speaking rate globally by interpolation respectively.

were fed to the acoustic predictor trained with the bottleneck features of the target speaker [21]. This inconsistency may degrade the similarity of converted speech. Another reason can be attributed to the duration conversion ability of our proposed method as introduced in the objective evaluations. Therefore, the prosody similarity of our proposed method was better than simply adjusting speaking rate globally.

#### D. Ablation tests

1) *Bottleneck features*: In order to investigate the necessity of using bottleneck features, we removed them and built SCENT models only utilizing source mel-spectrograms as inputs. Objective evaluation results of MCD and  $F_0$  RMSE on test set are presented in TABLE V. It can be found that removing bottleneck features led to higher  $F_0$  RMSE and MCD on test set. Listening to the converted audio samples without using bottleneck features, we found they suffered from serious mispronunciation problem. The bottleneck features extracted by an ASR model contain high-level and linguistic-related information. The experimental results indicate that they were essential for achieving stable voice conversion results in our proposed method.

2) *Attention module*: The attention module in a SCENT model helps to achieve the alignment between input and output feature sequences at the training stage and to predict target durations at the conversion stage. In order to investigate how the attention module contributing to the overall performance of our proposed method, we modified the SCENT model to a frame-by-frame transformation model without attention mechanism for comparison. Once the attention module was removed, the LSTM layer with attention in the decoder became a plain uni-directional LSTM. In order to get frame aligned sequence pairs for model training, the input sequences were wrapped towards the target ones using DTW algorithm and

TABLE VII  
THE RESULTS OF PREFERENCE TESTS ON SIMILARITY AMONG PROPOSED METHODS WITH AND WITHOUT THE ATTENTION MODULE.

	w/o-att (%)	i-w/o att (%)	w-att (%)	N/P (%)	$p$
F-M	33.0	<b>58.5</b>	-	8.5	$1.31 \times 10^{-4}$
	-	21.0	<b>67.5</b>	11.5	$< 1 \times 10^{-9}$
M-F	17.5	<b>76.0</b>	-	6.5	$< 1 \times 10^{-9}$
	-	24.0	<b>66.5</b>	9.5	$< 1 \times 10^{-9}$

“ $p$ ” represents  $p$  value of  $t$ -test. “N/P” denotes no preference. “F-M” and “M-F” represent female-to-male and male-to-female conversion respectively.

MCCs features. The other parts of the SCENT model were kept unchanged.

Our experiments compared three methods, including the proposed method with attention (w-att), the proposed method without attention (w/o-att) and the proposed method without attention but using source interpolation at conversion time (i-w/o-att). TABLE VI shows the MCDs and  $F_0$  RMSEs of these three methods. We can see that the prediction errors increased in the absence of the attention module.

Furthermore, a group of preference tests were conducted to compare the subjective performance of these three methods. Because the most significant advantage of our proposed method was similarity as discussed in Section IV-C2, the preference tests focused on the similarity aspect of converted speech. Ten native listeners were involved in evaluation and the experiment results are presented in TABLE VII. This table shows that the strategy of global speaking rate adjustment by source interpolation can improve the similarity of converted speech in both conversion pairs. The proposed method with attention module outperformed the method without attention but using source interpolation. These results further confirmed the effectiveness of the attention module.

#### E. Discussions

As discussed in Section II, directly implementing seq2seq models at utterance level is difficult for the voice conversion task. The input and output sequences in voice conversion are composed of frame-level features and relatively long thus it is a challenge for the attention mechanism to search for the correct hidden entries to pay attention to. Once there are abnormal skips or repetitions in the sequence of attention probabilities, mistakes of converted speech may occur.

These difficulties are considered when designing the SCENT model. In order to improve attention stability, the techniques of forward attention and adding location features are used when calculating attention probabilities. The bottleneck features can also provide linguistic-related information to help the attention-based alignment between input and output feature sequences. However, errors still can not be completely avoided in the converted speech. Additional 100 non-parallel utterances of both speakers out of the dataset used for previous experiments were adopted for error analysis. The utterances of the male speaker contained 2747 phonemes, while the utterances

of the female speaker had 2538 phonemes. We conducted male-to-female and female-to-male conversions for these utterances and identified different categories of conversion errors subjectively. In male-to-female conversion, there were 1 skipping phoneme error, 2 completion prediction errors, 34 phoneme pronunciation errors, 31 tone defects and 10 phoneme quality defects. In female-to-male conversion, there were 19 phoneme pronunciation errors, 20 tone defects and 17 phoneme quality defects.

Several reasons may lead to these errors. First, the proposed model is complex and needs to be trained in a data-driven way. Therefore, the insufficiency of training data may cause the model's lack of generalization ability when dealing with unseen utterances. Also, the extracted bottleneck features may also be misleading due to the accuracy limitation of the ASR model. To further reduce conversion errors and to produce more reliable conversion results using seq2seq models will be an important task of our future work.

## V. CONCLUSION

This paper presents SCENT, a sequence-to-sequence neural network, for acoustic modeling in voice conversion. Mel-spectrograms are used as acoustic features. Bottleneck features extracted by an ASR model are taken as additional linguistic-related descriptions and are concatenated with the source acoustic features as network inputs. Taking advantage of the attention mechanism, the SCENT model does not rely on the preprocessing of DTW alignment and the duration conversion can be achieved simultaneously. Finally, the converted acoustic features are passed through a WaveNet vocoder to reconstruct speech waveforms. Objective and subjective experimental results demonstrated the superiority of our proposed method compared with baseline methods, especially in durational aspect. Ablation tests further proved the necessity of appending bottleneck features and the importance of the attention module. To investigate the influence of training set size on the performance of our proposed method and to reduce conversion errors by improving attention calculation will be our work in the future.

## REFERENCES

- [1] D. G. Childers, B. Yegnanarayana, and K. Wu, "Voice conversion: Factors responsible for quality," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 1985, pp. 748–751.
- [2] D. G. Childers, K. Wu, D. M. Hicks, and B. Yegnanarayana, "Voice conversion," *Speech Communication*, vol. 8, no. 2, pp. 147–158, 1989.
- [3] A. Kain, "Spectral voice conversion for text-to-speech synthesis," *Proc. ICASSP-1998*, vol. 1, pp. 285–288 vol.1, 1998.
- [4] L. M. Arslan, "Speaker transformation algorithm using segmental codebooks (stasc) 1," *Speech Communication*, vol. 28, no. 3, pp. 211–226, 1999.
- [5] M. Müller, "Dynamic time warping," *Information retrieval for music and motion*, pp. 69–84, 2007.
- [6] 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.
- [7] S. Desai, E. V. Raghavendra, B. Yegnanarayana, A. W. Black, and K. Prahallad, "Voice conversion using artificial neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2009, pp. 3893–3896.
- [8] S. Desai, A. W. Black, B. Yegnanarayana, and K. Prahallad, "Spectral mapping using artificial neural networks for voice conversion," *IEEE Transactions on Audio Speech and Language Processing*, vol. 18, no. 5, pp. 954–964, 2010.
- [9] D. M. Titterton, A. F. M. Smith, and U. E. Makov, *Statistical analysis of finite mixture distributions*. Wiley, 1985.
- [10] K. Hornik, "Multilayer feedforward neural networks are universal approximators," *Neural Networks*, vol. 2, 1989.
- [11] R. H. Laskar, D. Chakrabarty, F. A. Talukdar, K. S. Rao, and K. Banerjee, "Comparing ANN and GMM in a voice conversion framework," *Applied Soft Computing Journal*, vol. 12, no. 11, pp. 3332–3342, 2012.
- [12] L. H. Chen, Z. H. Ling, L. J. Liu, and L. R. Dai, "Voice conversion using deep neural networks with layer-wise generative training," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 22, no. 12, pp. 1859–1872, 2014.
- [13] L. Sun, S. Kang, K. Li, and H. Meng, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4869–4873.
- [14] T. Nakashika, T. Takiguchi, and Y. Ariki, "Voice conversion using RNN pre-trained by recurrent temporal restricted boltzmann machines," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 3, pp. 580–587, 2015.
- [15] J. Lai, B. Chen, T. Tan, S. Tong, and K. Yu, "Phone-aware LSTM-RNN for voice conversion," in *IEEE International Conference on Signal Processing (ICSP)*, 2016, pp. 177–182.
- [16] S. H. Mohammadi and A. Kain, "An overview of voice conversion systems," *Speech Communication*, vol. 88, pp. 65–82, 2017.
- [17] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Neural Information Processing Systems*, pp. 3104–3112, 2014.
- [18] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *Empirical Methods in Natural Language Processing*, pp. 1724–1734, 2014.
- [19] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *International Conference on Learning Representations*, 2015.
- [20] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *Empirical Methods in Natural Language Processing*, pp. 1412–1421, 2015.
- [21] L. J. Liu, Z. H. Ling, and L. R. Dai, "WaveNet vocoder with limited training data for voice conversion," *Conference of the International Speech Communication Association, INTERSPEECH*, 2018.
- [22] Y. Wang, R. J. Skerryryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio et al., "Tacotron: Towards end-to-end speech synthesis," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2017, pp. 4006–4010.
- [23] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. J. Skerryryan et al., "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4779–4783.
- [24] W. Ping, K. Peng, A. Gibiansky, S. O. Arikan, A. Kannan, S. Narang, J. Raiman, and J. P. Miller, "Deep voice 3: 2000-speaker neural text-to-speech," *International Conference on Learning Representations*, 2018.
- [25] H. Tachibana, K. Uenoyama, and S. Aihara, "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4874–4878.
- [26] M. V. Ramos, "Voice conversion with deep learning," Master's Thesis, Instituto Superior Técnico, 10 2016.
- [27] T. Kaneko, H. Kameoka, K. Hiramatsu, and K. Kashino, "Sequence-to-sequence voice conversion with similarity metric learned using generative adversarial networks," in *Conference of the International Speech Communication Association*, 2017, pp. 1283–1287.
- [28] H. Miyoshi, Y. Saito, S. Takamichi, and H. Saruwatari, "Voice conversion using sequence-to-sequence learning of context posterior probabilities," *arXiv preprint arXiv:1704.02360*, 2017.
- [29] A. V. Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," *arXiv: Sound*, p. 125, 2016.
- [30] K. Kobayashi, T. Hayashi, A. Tamamori, and T. Toda, "Statistical voice conversion with WaveNet-based waveform generation," in *Proceedings*

- of the Annual Conference of the International Speech Communication Association, *INTERSPEECH*, 2017, pp. 1138–1142.
- [31] K. H. Jumpei Niwa, Takenori Yoshimura, Y. N. Keiichiro Oura, and K. Tokuda, “Statistical voice conversion based on WaveNet,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5289–5293.
  - [32] M. Morise, F. Yokomori, and K. Ozawa, “World: A vocoder-based high-quality speech synthesis system for real-time applications,” *Ieice Transactions on Information and Systems*, vol. 99, no. 7, pp. 1877–1884, 2016.
  - [33] H. Kawahara, I. Masuda-Katsuse, and A. D. Cheveign, “Restructuring speech representations using a pitch-adaptive timefrequency smoothing and an instantaneous-frequency-based f0 extraction: Possible role of a repetitive structure in sounds 1,” *Speech Communication*, vol. 27, no. 34, pp. 187–207, 1999.
  - [34] X. Wang, J. Lorenzo-Trueba, S. Takaki, L. Juvela, and J. Yamagishi, “A comparison of recent waveform generation and acoustic modeling methods for neural-network-based speech synthesis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4804–4808.
  - [35] Y. Ai, H. C. Wu, and Z. H. Ling, “SampleRNN-based neural vocoder for statistical parametric speech synthesis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5659–5663.
  - [36] S. Hchreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
  - [37] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 4960–4964.
  - [38] J. Lei Ba, J. R. Kiros, and G. E. Hinton, “Layer Normalization,” *arXiv preprint arXiv:1067.06450*, 2016.
  - [39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, 2017, pp. 6000–6010.
  - [40] J. X. Zhang, Z. H. Ling, and L. R. Dai, “Forward attention in sequence-to-sequence acoustic modeling for speech synthesis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4789–4793.
  - [41] C. M. Bishop, “Mixture density networks,” 1994.
  - [42] M. B. Christopher, *Pattern Recognition and Machine Learning*. Springer-Verlag New York, 2016.
  - [43] B. Xu, N. Wang, T. Chen, and M. Li, “Empirical evaluation of rectified activations in convolutional network,” *arXiv preprint arXiv:1050.00853*, 2015.
  - [44] D. Krueger, T. Maharaj, J. Kramar, M. Pezeshki, N. Ballas, N. R. Ke, A. Goyal, Y. Bengio, A. C. Courville, and C. Pal, “Zoneout: Regularizing RNNs by randomly preserving hidden activations,” *International Conference on Learning Representations*, 2017.
  - [45] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *Computer Science*, 2014.
  - [46] D. T. Chappell and J. H. L. Hansen, “Speaker-specific pitch contour modeling and modification,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, 1998, pp. 885–888.
  - [47] Y. Ohtani, T. Toda, H. Saruwatari, and K. Shikano, “Maximum likelihood voice conversion based on GMM with straight mixed excitation,” in *Proc. ICSLP*, 2006, pp. 2266–2269.
  - [48] Z. Wu, O. Watts, and S. King, “Merlin: An open source neural network speech synthesis system,” in *Proc. 9th ISCA Speech Synthesis Workshop (SSW9)*, 2016.