EXTRACTING STRUCTURAL SPECTRAL FEATURES USING WHAT-WHERE AUTO-ENCODERS FOR STATISTICAL PARAMETRIC SPEECH SYNTHESIS

Ya-Jun Hu, Zhen-Hua Ling, Li-Rong Dai

National Engineering Laboratory of Speech and Language Information Processing
University of Science and Technology of China, Hefei, P.R.China
hyj15475@mail.ustc.edu.cn, {zhling,lrdai}@ustc.edu.cn

ABSTRACT

This paper presents a method to extract structural spectral features from spectral envelopes using what-where auto-encoders (WWAE) for statistical parametric speech synthesis (SPSS). A WWAE is constructed by concatenating a convolutional net for input encoding and a deconvolutional net for reconstruction. The output values of the max-pooling layer in the encoder and the positions of the max-pooling switches are utilized as the *what* and *where* features respectively. Considering the intrinsic formant structures in the spectral envelopes of voiced speech frames, the WWAE model is adopted in this paper to detect, locate, and reconstruct the formants and other local structures in spectral envelopes. Here, the *what* and *where* features describe the prominences and positions of specific local spectral structures within a pooling frequency window. Then, the extracted *what* and *where* features are modeled as separate streams under the hidden Markov model (HMM)-based SPSS framework. Experimental results show that the speech synthesis system built using our proposed spectral features can produce synthetic speech with sharper formant structures and better naturalness than the systems using mel-cepstra and conventional auto-encoder-based spectral features.

Index Terms— speech synthesis, convolution neural network, what-where auto-encoder, spectral envelope, hidden Markov model

1. INTRODUCTION

Hidden Markov model (HMM) based statistical parametric speech synthesis (SPSS) [1] is one of the most popular methods for speech synthesis nowadays. This method is able to synthesize highly intelligible and smooth speech, and has various advantages such as compact footprint and the flexibility to control the characteristics of synthetic speech. However, this method has a tendency to over-smooth the spectral envelopes of synthetic speech. In recent years, neural network based methods have emerged as another promising way for SPSS. The methods of applying deep learning techniques to improve the acoustic models and describing the relationship between input texts and the corresponding acoustic features have been well investigated in the past few years [2, 3, 4, 5]. However, the over-smoothing problem has not been fully addressed due to the statistical averaging nature of SPSS.

On the other hand, the performance of solving a machine learning task also strongly relies on the strategy of data representation it adopts [6]. Neural-network-based spectral feature representations have been studied for speech synthesis. Their idea was to adopt the hidden representations in an auto-encoder (AE) [7] or a deep belief network (DBN) [8] as spectral features for acoustic modeling, which showed better performance than conventional spectral features, such as mel-cepstra and line spectral pairs (LSPs). However, none of these features paid specific attention to the local spectral structures, e.g., formants, of voiced speech frames, which are considered to be essential for speech perception. The synthesized formant structures are usually over-smoothed, which is one of the main reasons causing the quality degradation of synthetic speech.

Convolution neural network (CNN) is one of the most popular models for image-related classification tasks [9] because of its strong ability to detect local structural features[10]. Zhao, et al. [11] proposed a novel CNN architecture called what-where auto-encoder (WWAE), which uses the output values (i.e., *what* features) and switch positions (i.e., *where* features) of max-pooling operation to encode input. This paper proposes to adopt this WWAE architecture as a spectral feature representation model, which is able to detect, locate and reconstruct the formants and other local structures in the spectral envelopes of voiced speech segments. Single dimensional convolution along the frequency axis is adopted in the WWAE. Here, the extracted *what* and *where* features represent the prominences and positions of local structures in the spectral envelopes and are used as the spectral features for acoustic modeling under HMM framework. The

This work is partially funded by the National Nature Science Foundation of China (Grant No.61273032), the CAS Strategic Priority Research Program (Grant No. XDB02070006), the National Key Research and Development Program of China (Grant No.2016YFB100130300) and the Fundamental Research Funds for the Central Universities (Grant No.WK2350000001).

what features are modeled by Gaussian distributions, and the *where* features are modeled as independent z-class random variables, where z is the pooling size. Maximum output probability criterion is adopted at synthesis time to predict these features for spectral envelope reconstruction.

2. METHODS

2.1. What-where auto-encoder (WWAE)

A one-layer WWAE [11] consists of a convolution operation and a pooling operation during encoding as shown in the left of Fig 1. For a WWAE with N hidden feature maps, the N filters compose the weight matrix of the WWAE. One-layer WWAEs can be stacked up to form a deep architecture.

In the encoding phase, the hidden feature maps are calculated as the output of the convolution between the input feature vector and the corresponding filters after a nonlinear activation function, written as

$$\boldsymbol{q}_i = g(\boldsymbol{o} * \boldsymbol{w}_i), \tag{1}$$

where $\boldsymbol{o}$ denotes the input feature vector, $\boldsymbol{q}_i$ represents the i-th hidden feature map, $\boldsymbol{w}_i$ represents the filter for calculating the i-th hidden feature map, $*$ is the convolution operation, and $g(x)$ is the activation function. In the pooling layer, a max-pooling operation with pooling size z is conducted for each hidden feature map as shown in the right of Fig. 1. In this operation, the max value in a pooling window is retained, which is called *what*, and the position of its corresponding switch is stored, which is called *where*. The *what* and *where* features are calculated as

$$p_{i,m} = \max_{n\in[0,z-1]} q_{i,(m-1)*z+n}, \tag{2}$$

$$s_{i,m} = \operatorname*{argmax}_{n\in[0,z-1]} q_{i,(m-1)*z+n}, \tag{3}$$

where $p_{i,m}$ and $s_{i,m}$ represent the *what* and *where* features of the m-th pooling window in the i-th hidden feature map, and $q_{i,j}$ is the j-th element of $\boldsymbol{q}_i$.

In the decoding phase, unpooling is conducted using the *what* and *where* features extracted above. As shown in the right of Fig. 1, in an unpooling operation, a z-dimension vector is generated for each pooling window where the *what* value is filled into the *where* position in this vector and the other elements of this vecotr are set to be 0. Given the unpooling output $\boldsymbol{q}'_i$, the input is reconstructed by the convolution between the hidden feature maps and the filters, written as

$$\boldsymbol{o}' = g'(\sum_i \boldsymbol{q}'_i * \boldsymbol{w}_i), \tag{4}$$

where $\boldsymbol{o}'$ is the reconstructed input feature vector and $g'(x)$ is the activation function for reconstruction.

The WWAE model parameters can be estimated by gradient descent method under a criterion of minimizing the square errors of reconstructing input features and hidden states [11].

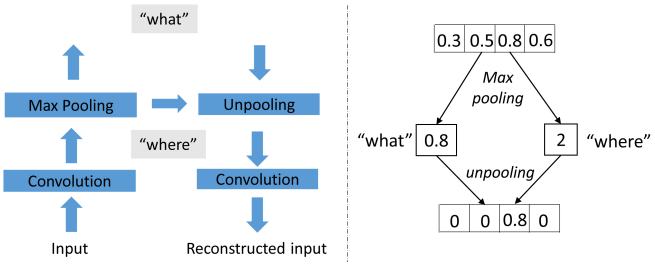

Fig. 1. Left: The structure of a one-layer WWAE. Right: An example of max-pooling and unpooling operations with pooling size $z = 4$.

2.2. WWAE-based spectral representation for SPSS

In this paper, we propose to adopt the *what* and *where* features derived from spectral envelopes using a WWAE as the spectral representations for SPSS. Here, the WWAE is set to be one layer and the convolution is only conducted along the frequency axis. As described in Section 2.1, *what* features $\boldsymbol{p}_i = \{p_{i,m}\}_m$ are the output values of the max-pooling operation and *where* features $\boldsymbol{s}_i = \{s_{i,m}\}_m$ are the positions of the corresponding max-pooling switches. Suppose the dimensionality of the input spectral envelope is D_i, the filter length is D_f, and the pooling size is z. Then, the dimensionality of $\boldsymbol{p}_i$ and $\boldsymbol{s}_i$ is $M = \lfloor (D_i - D_f + 1)/z \rfloor$ for each of the N feature maps. $\boldsymbol{p}_i$ are real-valued and $\boldsymbol{s}_i$ contain integers between 0 and $z-1$. Given a training set for SPSS, the spectral envelopes extracted by STRAIGHT vocoder [12] at voiced frames are used to estimated the parameters of the WWAE. The training algorithm in [11] is followed where only the reconstruction error of input features is considered in the criterion. Then, $\boldsymbol{p}_i$ and $\boldsymbol{s}_i$ features of each voiced frame can be calculated using the trained WWAE model.

We expect that the extracted $\boldsymbol{p}_i$ and $\boldsymbol{s}_i$ can represent the prominences and positions of local spectral structures within a pooling frequency window. Therefore, averaging $\boldsymbol{p}_i$ can be considered as calculating the average prominence of the local spectral structures in a pooling window, while predicting $\boldsymbol{s}_i$ is to find the most popular positions of the local spectral structures within a pooling window. We expect to reduce the over-smoothing effect on the formants and other local spectral structures of synthetic speech by decomposing spectral envelopes into *what* and *where* representations and modeling them respectively.

The HMM-based SPSS framework is followed in this paper to build a speech synthesis system using the *what* and *where* features extracted above. First, a conventional HMM-based SPSS system, i.e., the baseline system, is constructed using mel-cepstra derived from STRAIGHT spectral envelopes as spectral features. Then, the *what* and *where* features extracted by WWAE are used to replace mel-cepstra to describe the spectral characteristics at each voiced context-dependent HMM state. Here, voiced context-dependent states

mean the clustered states describing voiced phonemes according to the built decision trees. After training context-dependent HMMs using mel-cepstra, a state alignment to the acoustic features is performed. For each voiced context-dependent state, the *what* and *where* features of all voiced frames belonging to this state can be determined according to the state alignment results. These two features are modeled in two streams separately. At each HMM state, the *what* features are modeled by a Gaussian distribution with diagonal covariance matrix and each dimension in the *where* features is modeled as a z-class discrete random variables.

At synthesis time, an HMM state sequence is first determined using the text analysis and duration prediction results. For each frame in the voiced states, the *what* and *where* features are predicted under maximum output probability criterion, i.e., the *what* features are generated as the Gaussian mean vector of current state and the *where* features are predicted to be the positions with maximum probabilities in the z-class discrete distributions. The generated *what* and *where* features of voiced frames are further converted into spectral envelopes using the decoding part of the WWAE. For each frame in the unvoiced states, the static mel-cepstra is also predicted under maximum output probability criterion, which is the static part of the Gaussian mean vector of current state. Then, spectral envelopes are recovered from the predicted mel-cepstra for unvoiced frames. Since non dynamic features are utilized, the predicted spectral envelopes are constant within each HMM state and are discontinuous at state boundaries. In our implementation, they are smoothed before being sent into the synthesizer using the algorithm of parameter generation with the constraints of dynamic features[8], in which the mean vectors for dynamic features and the diagonal covariance matrices are calculated from the spectral envelopes assigned to each HMM state in the training set.

3. EXPERIMENTS

3.1. Experimental conditions

The data of the female US English speaker SLT in the CMU ARCTIC database (`http://festvox.org/cmu_arctic/`) was used in our experiments. The waveforms were recorded in 16kHz/16bit format. One thousand utterances in the database were used for system training and the remaining 132 sentences were used as a test set.

In the HMM-based baseline system, 41-order mel-cepstra (including the 0-th order for frame power) and F0 along with their dynamic components were used as acoustic features. The spectrum part was modeled by a Gaussian distribution with diagonal covariance matrix, and the F0 part was modeled by a multi-space probability distribution (MSD)[13].

The dimension of the spectral envelopes used in our experiments was $D_i = 513$ due to the FFT length of 1024 during STRAIGHT analysis. The logarithmic spectral envelopes of all voiced frames in the training set were used as input to train a WWAE model. Sigmoid activation function was used in (1) and linear activation function was used in (4) in this model. In a WWAE model, filter length D_f represents the width of the local structures that we want to detect. In this paper, the filters are expected to detect the formant-like structures and some other local spectral structures contained in the spectral envelopes of voiced speech. Here, the filter length was heuristically set to be $D_f = 34$ in our implementation, which covered about 530 Hz. On the other hand, the pooling size determines the downsampling rate of the convolution output and how many spectral details can be preserved after reconstruction. In our experiments, the pooling size was set as $z = 20$, corresponding to about 300 Hz. Therefore, the dimension of the extracted *what* and *where* features was 24 for each feature map according to the introduction in Section 2.2. The number of feature maps represents how many local structures we want to detect, which was investigated in our experiments.

Table 1. Log spectral distortions of analysis-by-synthesis using WWAEs with different numbers of feature maps (N).

N	5	10	20	30	40	50
LSD (dB)	1.93	1.68	1.68	1.69	1.73	1.74

3.2. Analysis-by-synthesis experiments

The WWAE models with different numbers of feature maps were trained and then compared in an analysis-by-synthesis experiment. In this experiment, the spectral envelopes of the voiced frames in the test set were transformed into *what* and *where* features and then converted back into spectral envelopes using the estimated WWAE models. The log spectral distortion between the original and reconstructed spectral envelopes in the test set were calculated and are shown in Table. 1. From this table, we can see that the LSD was large when there were not enough feature maps, e.g., $N = 5$. On the other hand, increasing the number of feature maps too much cannot improve the performance of modeling spectral envelopes. This implies that the number of local structures appearing in the spectral envelopes should be limited.

To better understand what have been learnt during the WWAE training, we plot the estimated filter weights of WWAEs with 10 feature maps and 50 feature maps in Fig. 2 and Fig. 3 respectively. The width of each subfigure, i.e., the filter length, was 34 as introduced above. We can see that most of the filters in Fig. 2 have formant-like or anti-formant-like shapes. While, the filters in Fig. 3 have smaller absolute values and more fluctuations. Most of them are noise-like and have no clear formant or anti-formant patterns. Considering its best performance in Table 1 and the explainable filter shapes, the WWAE model with 10 feature maps was adopted in our following experiments.

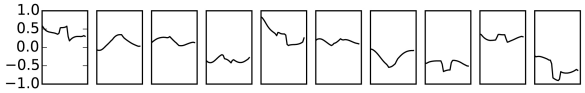

Fig. 2. Estimated filter weights with 10 feature maps.

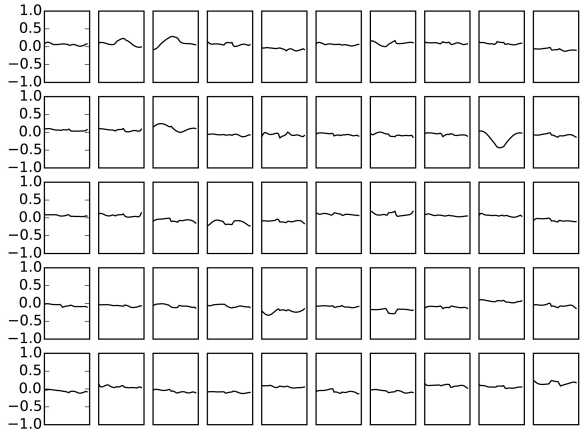

Fig. 3. Estimated filter weights with 50 feature maps.

3.3. Speech synthesis experiments

In the speech synthesis experiments, three systems using different spectral features were constructed and compared.

- *Baseline*: the baseline HMM-based system using mel-cepstra as spectral feature.
- *AE*: this system utilized hidden representations of an auto-encoder (AE) as spectral features. There were two hidden layers and 1024 units in each layer in the AE. A Gaussian distribution with diagonal covariance matrix was adopted to model the AE-based spectral features at each HMM state.
- *WWAE*: this system utilized the *what* and *where* features extracted by the WWAE model with 10 feature maps as spectral features. The system construction followed the method introduced in Section 2.2.

These three systems shared the same decision trees, F0 models, and duration models which was estimated by the *Baseline* system.

A MUSHRA (MUltiple Stimuli with Hidden Reference and Anchor) test [14] was conducted to compare the naturalness of these three systems with natural recordings as references.[1] Twenty sentences synthesized by the three systems were evaluated by 30 English native listeners on the crowdsourcing platform of Amazon Mechanical Turk (`http://www.mturk.com`) with anti-cheating considerations [15]. The average naturalness scores of these three systems are shown in Table. 2. Our proposed method outperformed *AE* and *Baseline* on naturalness score.

[1]Demos of synthetic speech can be found at `http://home.ustc.edu.cn/~hyj15475/ICASSP2017_WWAE/demo.html`.

Table 2. MUSHRA evaluation results of the constructed systems using different spectral features. The differences between *Baseline* and *WWAE*, and between *AE* and *WWAE* are significant ($p < 0.01$ in paired t-test).

System	*Baseline*	*AE*	*WWAE*
Naturalness score	39.7	40.6	44.3

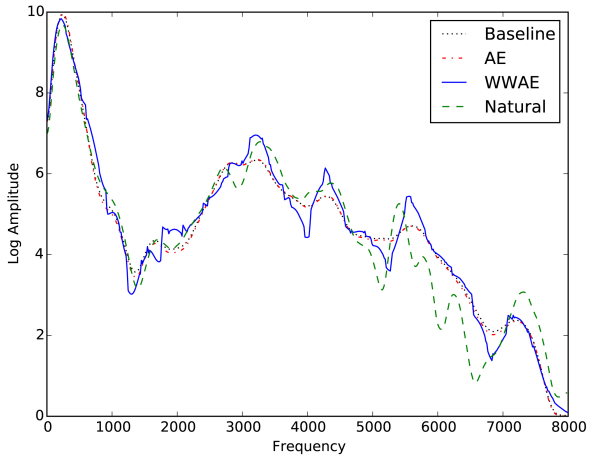

Fig. 4. The spectral envelopes generated by different systems for a voiced HMM state and a natural sample.

Fig. 4 shows the spectral envelopes generated by different systems for a voiced HMM state with a natural sample as a reference. We can see that the spectral envelope predicted by *WWAE* retains the local formant and anti-formant structures better than the other two systems.

4. CONCLUSIONS

We have proposed to adopt a what-where auto-encoder (WWAE) as a spectral feature extractor for SPSS. The WWAE is able to detect, locate and reconstruct local structures in spectral envelopes. Experimental results show that modeling and predicting the *what* and *where* features separately can relieve the averaging effect on generated spectral envelopes.

Further investigation is still necessary based on current results. First, the architecture of WWAE could be improved, e.g., using a deeper WWAE architecture and two dimensional convolution along time and frequency axes. Second, the representation of *where* features could be improved to make it easier for acoustic modeling. Third, better acoustic models could be applied, such as deep neural networks (DNNs), recurrent neural networks (RNNs). Fourth, our proposed method should be evaluated when integrating other spectral enhancement methods such as parameter generation considering global variance (GV)[16].

5. REFERENCES

[1] Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi, and Tadashi Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," in *Proc. Eurospeech*, 1999, vol. 6, pp. 2347–2350.

[2] Zhen-Hua Ling, Li Deng, and Dong Yu, "Modeling spectral envelopes using restricted Boltzmann machines and deep belief networks for statistical parametric speech synthesis," *IEEE transactions on audio, speech, and language processing*, vol. 21, no. 9-10, pp. 2129–2139, 2013.

[3] Shiyin Kang, Xiaojun Qian, and Helen Meng, "Multi-distribution deep belief network for speech synthesis," in *Proc. ICASSP*. IEEE, 2013, pp. 8012–8016.

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

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

[6] Yoshua Bengio, Aaron Courville, and Pascal Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[7] Shinji Takaki, SangJin Kim, Junichi Yamagishi, and JongJin Kim, *Multiple Feed-forward Deep Neural Networks for Statistical Parametric Speech Synthesis*, pp. 2242–2246, International Speech Communication Association, 2015.

[8] Y. J. Hu and Z. H. Ling, "DBN-based spectral feature representation for statistical parametric speech synthesis," *IEEE Signal Processing Letters*, vol. 23, no. 3, pp. 321–325, March 2016.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[10] Matthew D Zeiler and Rob Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*. Springer, 2014, pp. 818–833.

[11] Junbo Zhao, Michael Mathieu, Ross Goroshin, and Yann LeCun, "Stacked what-where auto-encoders," arXiv preprint `http://arxiv.org/abs/1506.02351v8`.

[12] Hideki Kawahara, Ikuyo Masuda-Katsuse, and Alain de Cheveigné, "Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech communication*, vol. 27, no. 3, pp. 187–207, 1999.

[13] Keiichi Tokuda, Takashi Masuko, Noboru Miyazaki, and Takao Kobayashi, "Hidden Markov models based on multi-space probability distribution for pitch pattern modeling," in *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*. IEEE, 1999, vol. 1, pp. 229–232.

[14] BS. 1534-1. Recommendation, ITUR, "Method for the subjective assessment of intermediate sound quality (MUSHRA)," *International Telecommunications Union, Geneva*, 2001.

[15] Sabine Buchholz and Javier Latorre, "Crowdsourcing preference tests, and how to detect cheating.," in *Proc. Interspeech*, 2011, pp. 3053–3056.

[16] T. Toda and K. Tokuda, "Speech paramter generation algorithm considering global variance for HMM-based speech synthesis," in *Proc. Interspeech*, 2005, pp. 2801–2804.