Experiments in Making VOCALOID Synthesis More Human-like Using Deep Learning

Michael Wilson[1,a] Pritish Chandna[2] Ryunosuke Daido[1,b] Yuji Hisaminato[1,c]

Abstract: Deep learning has recently been used to improve the results of many speech-related tasks. We applied deep learning to VOCALOID(TM), a singing voice synthesizer which uses concatenative synthesis, with the goal of making the synthesized sound more human-like. Previous work in this area includes using Hidden Markov Models (HMMs) to model the prosodic features of a specific singer or style, and using iterative parameter estimation to mimic target human singing. We focused on methods which work directly on audio data and audio features which can be automatically extracted, with no special markup or target singing required. We report the results of several experiments with various models and parameterizations, and suggest avenues for further research.

Keywords: Deep Learning, Singing Voice

1. Introduction

Our ultimate aim is to make human-like singing synthesis, and to quantify what that means. Although many singing synthesis systems have been proposed [1] [3], we do not think that there has been a system which produces satisfactory human-like singing. Current mainstream singing synthesizers can be broadly divided into unit concatenation systems such as VOCALOID [5] and statistical parametric modeling systems such as Sinsy [9]. VOCALOID, by avoiding low-dimensional parameterization and statistical processing, has an advantage in local naturalness of sound. On the other hand, statistical methods have an advantage in generating various sounds based on context information.

Recently, deep learning has produced significant improvements in voice-related tasks. In singing voice synthesis, [8] reported an improvement in naturalness by replacing Hidden Markov Models (HMMs) with Deep Neural Networks (DNNs). DNNs can be considered a type of statistical modeling. However, due to the wide variety of DNN architectures and training methods the best deep learning method for synthesizing singing has not been clarified. Furthermore, a wide variety of voice parameterizations are currently being considered. In [12], speech synthesis using deep learning frameworks did not achieve the best performance when using the traditional source-filter model. This implies that the parameterization should not be assumed when using DNNs. For example, [13] forgoes the standard concept of parameterization and models waveforms directly to produce speech synthesis.

We are using deep learning to explore singing. As discussed above, there is a lot of diversity in deep learning frameworks for handling voice and singing synthesis. Therefore, we decided not to start with conventional frameworks such as inputting score information and outputting singing voice audio. In this paper we report the results of a series of early experiments which apply deep learning to human singing and VOCALOID synthesis in order to find compact representations, compare them, and adapt between them.

The rest of the paper is organized as follows. Section 2 describes general methods, data, and tools. Section 3 describes five different experiments. Section 4 provides a brief discussion of the results and suggestions for future work. Section 5 contains concluding remarks and acknowledgments.

2. Methods

This study takes the approach of obtaining or generating audio data, performing transforms or feature extraction, then using existing GPU-accelerated deep learning frameworks to build neural networks which operate on the transformed data or extracted features. When possible and relevant, objective measures of performance such as validation accuracy are computed. Formal subjective listening tests are not performed in this study. In contrast to other work based on HMMs [11] or iterative parameter estimation [7] which modify synthesizer controls, this work attempts to operate directly on audio data and features extracted from audio data.

2.1 Audio Data

Natural human singing audio data ("Human Data") was gathered from the following sources:

- Recordings provided by Yamaha Corporation, consisting of 3.42 hours of amateur and professional Japanese singing in 168 files and 2.53 hours of professional English singing in 70 files

[1] Yamaha Corporation
[2] Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain
[a] michael.wilson@music.yamaha.com
[b] ryunosuke.daido@music.yamaha.com
[c] yuji.hisaminato@music.yamaha.com

- Recordings provided by the Universitat Pompeu Fabra Music Technology Group, consisting of 1.47 hours of English singing in 59 files and 0.24 hours of Spanish singing in 8 files
- MIR-1K dataset[*1], consisting of 2.15 hours of Chinese amateur singing in 110 files

The recordings include both male and female singers and were recorded in a variety of acoustic environments. The recordings were manually preprocessed to remove periods of silence over two seconds in length, and had an equal loudness filter applied.

VOCALOID synthesized singing data ("VOCALOID Data") was generated using the VOCALODUCER [6] system, which takes simple text lyrics as input. The Tatoeba text corpus[*2] was used to generate lyrics for each song. The lyrics for each song were composed of four sentences from the corpus in the target language, using up to 64 characters from each sentence. 797 English songs were synthesized with a single male VOCALOID voice, resulting in 4.5 hours of data. 779 Japanese songs were synthesized, split roughly evenly between four VOCALOID male voices and four female voices, resulting in 4.4 hours of data. The songs were evenly split among five different VOCALODUCER styles.

A parallel corpus of human singing and VOCALOID singing ("Parallel Data") was also prepared. The human part of the corpus consists of the original recordings that were used to create one female and one male English VOCALOID voice. Synthesizer control parameters which matched the pitch, phonemes, and overall length of the original recordings were automatically generated. These control parameters were run through the VOCALOID synthesis engine to generate corresponding VOCALOID resynthesis samples.

All audio data was converted to 16-bit monaural at 44.1 kHz sampling rate if not already in that format.

2.2 Features

Magnitude and phase spectra were calculated from the audio data using 1024-sample FFTs, Hanning window, and 256 sample hop size (75% overlap). The magnitude spectra were normalized by the hop size. Fundamental frequency, loudness, and MFCCs were extracted from each of the 1024-sample frames using Essentia [*3]. Finally, the frames were combined into batches of 30 or 513 consecutive frames to capture more time context (174.15 ms or 2977.95 ms, respectively), with 50% overlap between consecutive batches. These decisions were made based on [2]. This representation produces two-dimensional arrays when the magnitude spectrum is used, allowing convolution in both time and frequency.

2.3 Execution environment

Experiments were conducted on a single computer with an Intel Core i7-6800K CPU, 64 GB of RAM, and one MSI Geforce GTX TITAN X GPU. Major software and versions used were as follows:

Table 1 Autoencoder models

	Model 1	Model 2	Model 3	Model 4
Parameters	15,420	307,830	2,109,969	22,890
Layers	Freq. Conv Deconv.	Freq. Conv Deconv.	LSTM	Freq. Conv LSTM Deconv.

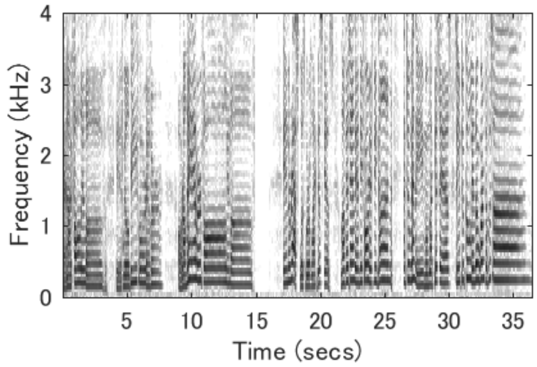

Fig. 1 Magnitude spectrum of one Human Data sample

- Ubuntu 10.04.5 LTS
- Nvidia driver version 367.57
- Cuda 8.0.44
- Python 2.7.6
- Theano 0.8.2
- Lasagne 0.2.dev1
- Essentia 2.1-beta3
- Numpy 1.11.1

3. Experiments and Results

This section describes five different experiments which were conducted.

3.1 Experiment 1: Autoencoder

The first experiment attempted to reduce the human singing voice to a minimal representation. An autoencoder with bottleneck [10] was used on the Human Data to attempt to extract relevant features. The hypothesis was that if the bottleneck could be made very small while maintaining high reconstruction quality then the representation in the bottleneck should contain a compact yet sufficient representation of the singing voice.

Several models were explored as shown in **Table 1**. The magnitude spectrum was used as the input feature and the Kullback-Leibler divergence was used as the cost function during training. Informal listening evaluation implies that LSTMs gave the best quality, but observing their state does not give much insight into what is learned by the network. Convolutional filters can be directly observed and applied individually to the input data after training is complete. One of the convolutional filters appeared to be influenced by the fundamental frequency of the voice as shown in **Fig. 1** and **Fig. 2**, but this assumption was not verified or pursued further.

3.2 Experiment 2: Classifier

The second experiment explored binary classification between VOCALOID and human singing using several network architectures. The hypothesis was that if a good classifier could be con-

[*1] https://sites.google.com/site/unvoicedsoundseparation/mir-1k
[*2] http://tatoeba.org/eng/downloads
[*3] http://essentia.upf.edu/

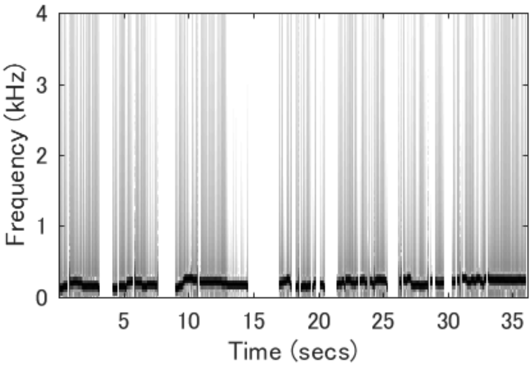

Fig. 2 One autoencoder filter applied to sample from Fig. 1

Table 2 Classification results for magnitude spectrum (1/2)

	Model A	Model B	Model C
Input	30,513	30,513	30,513
Conv	50 filters Shape=(1,513)	50 filters Shape=(10,513)	50 filters Shape=(25,513)
FC1	512 units Dropout=50%	1024 units Dropout=50%	256 units Dropout=50%
FC2	256 units Dropout=50%	512 units Dropout=50%	N/A
Softmax	2 units Dropout=50%	2 units Dropout=50%	2 units Dropout=50%
Valid	63.63%	75.83%	79.01%
Test	65.21%	75.07%	77.72%

Table 3 Classification results for magnitude spectrum (2/2)

	Model D	Model E	Model F
Input	30,513	30,513	30,513
Conv	50 filters Shape=(25,513)	50 filters Shape=(15,513)	50 filters Shape=(15,513)
FC1	512 units Dropout=50%	512 units Dropout=50%	1024 units Dropout=50%
FC2	256 units Dropout=50%	256 units Dropout=50%	512 units Dropout=50%
Softmax	2 units Dropout=50%	2 units Dropout=50%	2 units Dropout=50%
Valid	91.13%	93.85%	94.09%
Test	89.92%	91.64%	93.63%

structed then it may give insight into the differences between VOCALOID and human singing, and may also be useful as a component in a system to make VOCALOID singing more like human singing.

This experiment used the VOCALOID Data dataset and, since this dataset includes only English and Japanese singing, the English and Japanese samples from the Human Data dataset. Using the magnitude spectrum, accuracy over 90% was achieved as shown in **Table 2** and **Table 3**. Classification accuracy of over 85% was achieved even when training was only done on extracted pitch and loudness information, as shown in **Table 4**.

3.3 Experiment 3: Generative Adversarial Network

The third experiment explored using Generative Adversarial Networks [4], which combine a generative network with a classifier. The goal of this experiment was to build a network which would modify singing voice audio synthesized by VOCALOID in order to make it more human-like.

In this experiment the trained autoencoder and classifier from the previous experiments were used as a starting point for the generative network and classifier respectively. The same datasets as in Experiment 2 were used. Using the magnitude spectrum as the input feature results in changes to the timbre but does not induce large perceptual changes in the sound. Using pitch and loudness as input features results in more noticeable changes but does not necessarily result in a human-like output sound according to informal listening evaluations. Fixing the classifier, that is, allowing only the autoencoder to change, results in a strong humming noise at around 3,600 Hz when the magnitude spectrum is used as the input feature. This implies that the classifier may use information in this frequency range to distinguish between human and VOCALOID singing voice sounds.

Table 4 Classification results for pitch and loudness contours

	Model X	Model Y	Model Z
Input	513,2	513,2	513,2
Conv1	32 filters Shape=(15,1)	32 filters Shape=(15,1) ReLU	32 filters Shape=(15,1) ReLU
Conv2	32 filters Shape=(15,1)	32 filters Shape=(15,1) ReLU	N/A
FC1	1024 units Dropout=50%	1024 units Dropout=50%	1024 units Dropout=50%
FC2	512 units Dropout=50%	512 units Dropout=50%	N/A
Softmax	2 units Dropout=50%	2 units Dropout=50%	2 units Dropout=50%
Valid	80.45%	86.05%	87.44%
Test	79.98%	88.04%	87.57%

3.4 Experiment 4: Synthesizing voice from only pitch data

The fourth experiment attempted to synthesize singing from very limited input data. This experiment used the Human Data dataset. It started with an autoencoder which was trained on magnitude spectrum data. Then an LSTM was applied to learn an encoding between pitch curve data and the hidden representation in the autoencoder. Once this encoding was learned, it was used to produce synthesis of a full spectrogram using only pitch curve information. According to informal listening evaluations the resulting sound was very noisy and not very human-like. However, some vocal sonorant and fricative sounds were perceivable.

3.5 Experiment 5: Parallel dataset

Finally, in the fifth experiment a simple neural network was trained on the Parallel Data dataset to learn a mapping between VOCALOID and a human voice. Applying this network to a song which was not in the training set resulted in subtle changes in loudness and timbre. Informal listening evaluations suggested that this did not result in a more human-like sound, and the idea was not pursued further.

4. Discussion

The five experiments conducted imply that some methods can be used to produce only timbre changes, only pitch changes, only subtle changes, or large changes. Each of the experiments suggests avenues for further study. In particular:

- Experiment 1 suggests that convolutional filters may be able to learn features such as fundamental frequency.
- Experiment 2, by virtue of the high classification accuracy, suggests that there are detectable differences between the human voice and VOCALOID synthesis used in the experiment.

However, because of the lack of formal evaluation and follow-up experiments at this early stage it is not possible to draw any strong conclusions yet. Extensions of each experiment could include:

- Extend experiment 1 by searching for network architectures which produce convolutional filters that map to commonly-used audio features.
- Extend experiment 2 by quantifying the difference between the human voice and VOCALOID synthesis using the classifier as a guide.
- Extend experiment 3 by experimenting with different combinations of input features, network architectures, and training methods and conduct formal subjective evaluations.
- Extend experiment 4 by conducting formal listening tests on singing voice samples generated by neural nets using various combinations of input features, in order to quantify what level of quality can be obtained with a given set of input features.
- Extend experiment 5 by conducting formal listening tests on the parallel corpus results to quantify how the changes affect the perception of the sound.

5. Conclusions

We have presented the results of applying deep learning techniques in several different ways to the magnitude spectrum, pitch, and loudness curves of human and synthesized singing voice audio. Although we did not quantify what natural singing is, several possible follow-up experiments were suggested. Future work will incorporate these experimental results while investigating new methods for making human-like singing.

Acknowledgments We are grateful to Professor J.-S. Roger Jang from the CSIE Dept, National Taiwan Univ, Taiwan for giving us permission to use the MIR-1K dataset in this work.

The Tatoeba corpus was used under the Creative Commons CC-BY license (https://creativecommons.org/licenses/by/2.0/)

References

[1] Bloothooft, G.: Synthesis of singing challenge (online), available from ⟨http://www.interspeech2007.org/Technical/detail.php?ses=TuC.SS⟩ (accessed 2017-02-06).

[2] Chandna, P.: Audio Source Separation Using Deep Neural Networks, Master's thesis, Universitat Pompeu Fabra (2016).

[3] d'Alessandro, C.: Singing Synthesis Challenge: Fill-In the Gap (online), available from ⟨https://chanter.limsi.fr/doku.php?id=sidebar#special_session_interspeech_2016_singing_synthesis_challenge_fill-in_the_gap⟩ (accessed 2017-02-06).

[4] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y.: Generative Adversarial Nets, *Advances in Neural Information Processing Systems 27*, Curran Associates, Inc., pp. 2672–2680 (online), available from ⟨http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf⟩ (2014).

[5] Kenmochi, H. and Ohshita, H.: Vocaloid - commercial singing synthesizer based on sample concatenation, *Proc. INTERSPEECH*, pp. 4009–4010 (2007).

[6] Miyaki, T.: VOCALODUCER(TM): A System to Generate ”VOCALOID SONG” Automatically, *IPSJ-EC2015*, pp. 245–246 (2015).

[7] Nakano, T. and Goto, M.: VocaListener: A Singing-to-Singing Synthesis System Based on Iterative Parameter Estimation, *Sound and Music Computing Conference*, Vol. 6 (2009).

[8] Nishimura, M., Hashimoto, K., Oura, K., Nankaku, Y. and Tokuda, K.: Singing Voice Synthesis Based on Deep Neural Networks, *Interspeech 2016*, pp. 2478–2482 (online), DOI: 10.21437/Interspeech.2016-1027 (2016).

[9] Oura, K., Mase, A., Yamada, T., Tokuda, K. and Goto, M.: Sinsy – An HMM-based singing voice synthesis system which can realize your wish ”I want this person to sing my song”, *MUS*, Vol. 2010-MUS-86, No. 1, pp. 1–8 (2010).

[10] Sainath, T. N., Kingsbury, B. and Ramabhadran, B.: Auto-Encoder Bottleneck Features Using Deep Belief Networks, *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4153–4156 (2012).

[11] Tachibana, M., Saino, K. and Hisaminato, Y.: Application of HMM-Based Speech Synthesis Techniques to a Singing Style Synthesis Job Plugin, *SP IPSJ-SLP*, Vol. 113, No. 366, pp. 123–128 (2013).

[12] Takaki, S. and Yamagishi, J.: A deep auto-encoder based low-dimensional feature extraction from FFT spectral envelopes for statistical parametric speech synthesis, *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5535–5539 (2016).

[13] van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A. W. and Kavukcuoglu, K.: WaveNet: A Generative Model for Raw Audio, *CoRR*, Vol. abs/1609.03499 (online), available from ⟨http://arxiv.org/abs/1609.03499⟩ (2016).