Audio Style Transfer for Accents

Shuby Deshpande[*]
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
shubhand@cs.cmu.edu

Mantek Singh Chadha[*]
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
mschadha@andrew.cmu.edu

Victoria Lin[*]
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15213
vlin2@andrew.cmu.edu

1 Introduction

Originally proposed in the visual domain, style transfer refers to the changing of an image or video to adopt the visual style of another image or video (for example, modifying a painting by Monet to look as though it had been painted by Van Gogh). As attempts at performing visual style transfer have become more successful with the use of deep learning models, parallel efforts have emerged in the audio domain. For the audio modality, style transfer constitutes changing the style of an audio sequence to sound as though it were produced by another audio source. Tasks often explored include changing music played by one instrument to sound as if it has been played by another or changing a spoken utterance to sound as if it were spoken by someone else (voice conversion).

In our final project, we are specifically interested in one aspect of voice conversion—accent transfer. That is, we aim to modify an utterance spoken in Accent A (American English) to sound as though it is spoken in Accent B (British English), with other vocal characteristics (e.g. timbre, pitch) remaining constant. We distinguish accent from other aspects of utterance delivery like emotion, tone, and emphasis in that the intent and meaning of the utterance remains exactly the same, as does the vocal quality of the speaker.

To accomplish this goal, we draw inspiration prior successful work using generative models and autoencoders to perform analogous style transfer tasks in the visual domain. We propose an architecture with separate ground truth-trained generators and discriminators, which emulates a GAN-like structure without the use of random noise to generate fake data.

Our work consists of the following contributions:

- A new dataset of approximately $30,000$ parallel spoken utterances from the New York Times in American and British English accents, synthesized from the Amazon Polly service.
- A novel model that takes as input a spoken utterance in American or British English and produces audio of the same utterance spoken in the target accent.
- A website allowing users to explore examples of real accent-transferred audio produced by our model.

[*]equal contribution

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

2 Related Work

While deep learning-based image style transfer has seen significant advances in recent years [Gatys et al., 2016, Chen et al., 2017, Luan et al., 2017, Li et al., 2017], analogous tasks in the audio domain remain less explored. In particular, many aspects of voice-based style transfer are still in their methodological nascence. Following an initial successful effort using CNNs to texturize a target sound [Grinstein et al., 2018], a large portion of the body of work in audio style transfer has focused on instrumental music (changing timbre or playing style) or voice conversion (i.e. changing one person's voice to another's). Within these applications, the current state-of-the-art makes use of image-inspired unsupervised architectures like variational autoencoders [Mor et al., 2018, Qian et al., 2019] and GANs [Gao et al., 2018] (and variants like CycleGAN [Huang et al., 2019]), as well as fully supervised learning augmented with synthesized data [Cífka et al., 2019]. The success of these models informed our decision to use a GAN-like structure in our final architecture.

The approach we took to construct our discriminator in particular was grounded in work by Jia et al. [2018], who used a text-independent voice sample to encode characteristics of a person's voice. This voice embedding can then be used with an input sequence to transfer the style of the person's voice to the content of the input sequence [Jain et al., 2018]. We used their work as our basis in constructing an "accent embedding" that is independent of speaker characteristics.

Prior to the rise of deep learning in style transfer, the primary strategy for realistic voice conversion used on Gaussian mixture models [Zhou et al., 2018]. By learning a mapping of voice features like spectral envelope, formants, and mel-cepstrum from a source speaker to a target speaker using statistical methods like maximum likelihood estimation, Gaussian mixture models enable one-to-one voice conversion and even, with the appropriate data, tasks like emotion transfer [Kawanami et al., 2003]. These methods persist in the modern literature, but they are augmented by deep methods; for example, Kobayashi et al. [2017] used a GMM to map formants and mel-cepstrum between a source and target speaker, then used WaveNet [Oord et al., 2016] as a vocoder to synthesize speaker-converted waveform samples. Based on this convention, we decided to generate mel-spectrograms rather than .wav files and to perform the mel-to-wav conversion using an existing deep vocoder.

3 Description

3.1 Datasets

In this subsection, we introduce the datasets used to train our discriminator and generator, respectively.

3.1.1 CMU Arctic

CMU Arctic is a database of 1150 English-language utterances compiled by Carnegie Mellon University's Language Technologies Institute. The text for the utterances is derived from Project Gutenberg. Each utterance is read by 7 speakers, all of whom were living in the United States at time of reading. Of the speakers, 4 speak US-accented English, 1 speaks Canadian-accented English, 1 speaks Scottish-accented English, and 1 speaks Indian-accented English. We used this dataset to pre-train our model to generate accent embeddings, as explained below.

3.1.2 Amazon Polly

Amazon Polly [Polly, 2020] is a text-to-speech (TTS) service that offers both traditional and more recent deep learning techniques to synthesize speech. To generate a syntactically diverse dataset, we queried a year's worth of news articles from the New York Times and created a set of unique words. We then synthesized corresponding audio samples in both the American accent (male) and the British accent (male). These serve as our input and target for the accent transfer task. For the purpose of this experiment, we restricted ourselves to synthesizing only single word audio samples for accent transfer. We processed the data by eliminating empty samples (usually occurring when a non-English word is queried) and non-parallel utterances (usually occurring when an API call fails).

layer name	output size	params
conv1	×	3×3, 3, stride 2, padding 1
conv2_x	×	3×3, 3, stride 2, padding 1
pool_x	×	max pool
conv_fc	×	768×1024
lstm	×	input 128, hidden 128, layers 3, bidirectional
lstm_fc1	×	2048×1024
lstm_fc2	×	1024× (128 × 128)
FLOPs		×

Table 1: Architecture for generator model

layer name	output size	params
conv1d	×	250, 128, stride 50
batch norm	×	momentum 0.1
pool	×	max, kernel 4, stride 4
conv1d	×	30, 128, stride 1
batch norm	×	momentum 0.1
pool	×	max, kernel 4, stride 4
conv1d	×	30, 256, stride 1
batch norm	×	momentum 0.1
pool	×	max, kernel 4, stride 4
conv1d	×	30, 512, stride 1
batch norm	×	momentum 0.1
pool	×	adaptive, 1
Linear	×	512 × 50
Linear	×	50 × 2
FLOPs		×

Table 2: Architecture for discriminator model

3.2 Generator

We now give a brief overview of an end-to-end model that we have designed to solve the task of accent transfer. The model is trained on data generated using the Amazon Polly service, details of which have been explained section 3.1.2. For simplicity and ease of experimentation, we only considered the scenario of performing single-word accent transfer. We synthesized mel-spectrograms from the audio samples (source and target accents), which we then used as the input data and target label respectively. The generator architecture and parameters are shown in detail in 1 and in the context of the wider model in 1.

3.2.1 Generator architecture

Figure 1 shows the specific parameters we are using for the end to end model. In essence, there are 2 Conv layers to process the spectrogram and generate an embedding, which the three BiLSTM layers then ingest. There is an interleaving of BatchNorm and ReLU activations after each Conv block. We then do a MaxPool operation to generate a reduced-sized embedding, which is then processed using 1 FC layer before forwarding to the BiLSTM layers. The processed embedding from the LSTM layer is then re-projected to the space of spectrograms using 2 FC layers (which are again interleaved with ReLU activations).

3.3 Discriminator

In order to transfer an accent style, our model must be able to differentiate between two accents. A well-known technique in deep learning is to represent an object as a high-dimensional embedding vector. As shown in [Jain et al., 2018], accent embeddings can be used to improve speech recognition tasks and speaker accent classification. Thus, for our model to successfully transfer between accents,

we would like to feed it the representation of a target accent, from which it should be able to synthesize new audio, hopefully in the target accent. There have been previous efforts in [Jia et al., 2018] to synthesize audio in a target person's voice using speaker embeddings. Our approach generalizes these embeddings for an entire accent profile.

3.3.1 Discriminator architecture

We tried two different models to generate high-quality embeddings for accents. As shown in Table 3.2.1 below, the first model used 1-dimensional convolutions on raw audio signal, which was further processed by 1-dimensional convolutional layers above. The rationale for this approach is that a 1-D CNN can work as an effective feature extractor and help with data augmentation by windowing the raw input signal, as suggested by [Abdoli et al., 2019]. This network generates its own features from the raw audio signal and is able to capitalize on the time-dependent nature of audio data.

The second model we tried uses 2D-convolutions, which are applied after the input audio signal is preprocessed by creating spectrograms. A spectrogram of an audio signal conveys the spectral energy density across time and frequency axes. The spectrogram can then be used as a rich image representation of the audio signal, allowing for image classification techniques to be used on the spectrogram to classify the underlying audio signal. In [Hershey et al., 2016], the authors experimented by using well-known CNN-based image classification models for the purpose of audio classification, including ResNet-50. Since our dataset is considerably smaller, after some experimentation, we adapted a ResNet-34 model to use as our discriminator model. This model is shown in the pipeline of our end-to-end model in Figure 1.

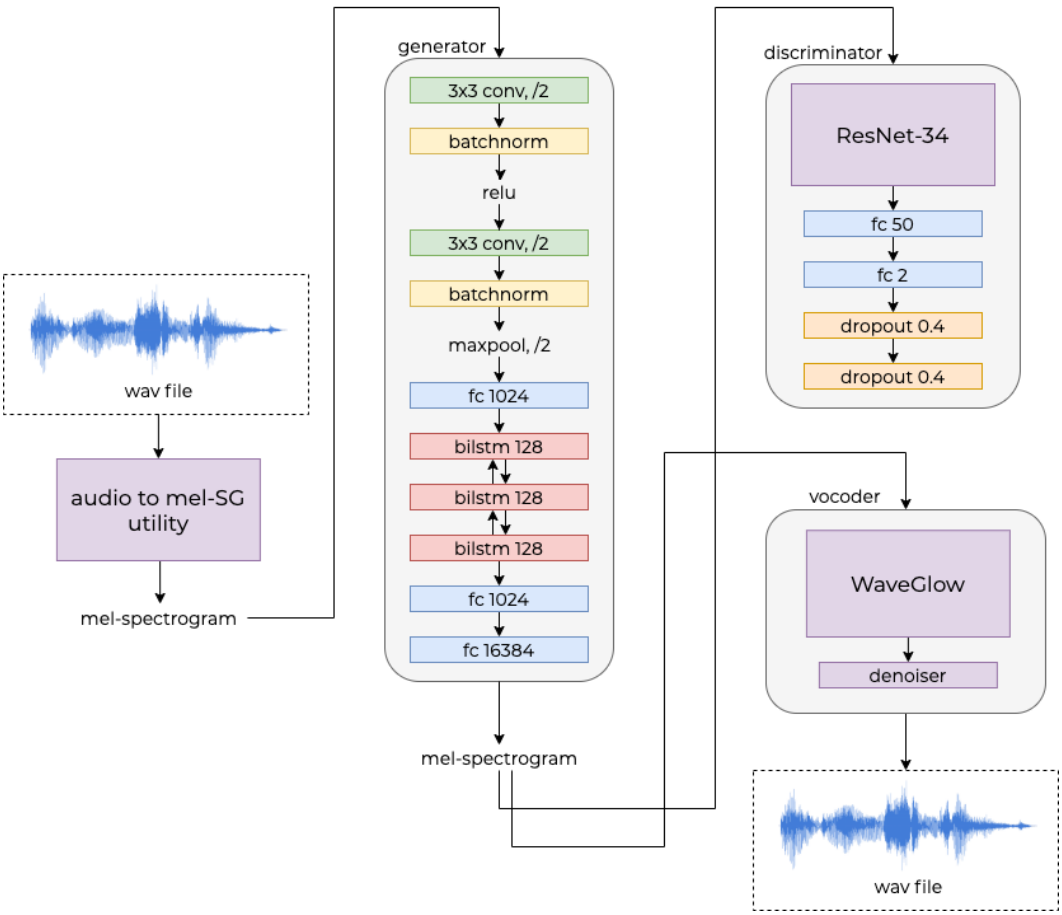

Figure 1: unbabel accent transfer model architecture

3.4 Vocoder

After generating mel-spectrograms representing the target audio, we passed the data into a vocoder to generate .wav audio. Perhaps the most conventional way of doing so is to use a fast generation algorithm like Griffin-Lim [Griffin and Lim, 1984]. Recently, however, deep vocoders like WaveNet [Oord et al., 2016] and WaveGlow [Prenger et al., 2019] have allowed for higher-quality synthesis of audio from mel-spectrograms. We selected WaveGlow as our vocoder due to its fast inference and added a denoiser to eliminate extreme frequencies, then integrated this component directly into our model pipeline.

4 Experiments

4.1 Generator

The final model is trained using an L1 loss against the target spectrogram, using the Adam optimizer. The full details of the hyperparameters we use are detailed in the appendix, along with links to the source code. Plots

4.2 Discriminator

To train the accent-embedding model, we used a weighted average of softmax cross-entropy loss and center loss. As demonstrated by [Qi and Su, 2017], contrastive center loss is able to improve performance on image classification tasks by learning a class center for each class, which enhances the discriminative power of learned features. We experimented with SGD and Adam optimizers. SGD appeared to have been stuck in local minima, and Adam performed comparatively better.

4.3 Vocoder

We evaluated audio reconstruction quality on both WaveNet and WaveGlow to determine the amount of reconstruction loss introduced by each model. We encoded raw WAV audio as mel-spectrograms for audio samples on the CMU Arctic dataset, then converted them back to WAV using both deep architectures. We also compared inference time.

5 Results

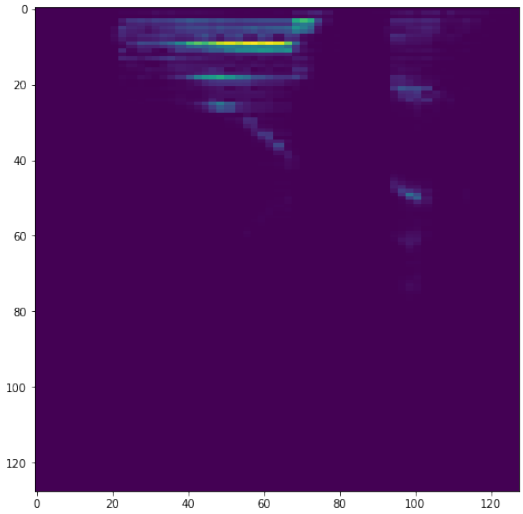

Figure 2: Source SG (American Accent)

The figures illustrate the transferred accent which was generated using the method described in the experiments section. As is evident, the ground truth and synthesized spectrograms are fairly similar

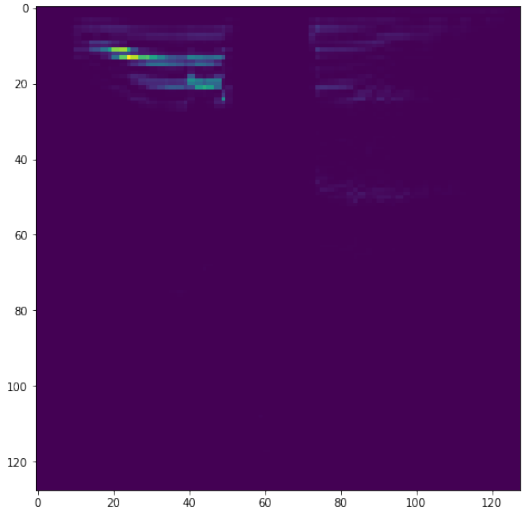

Figure 3: Target SG (British Accent)

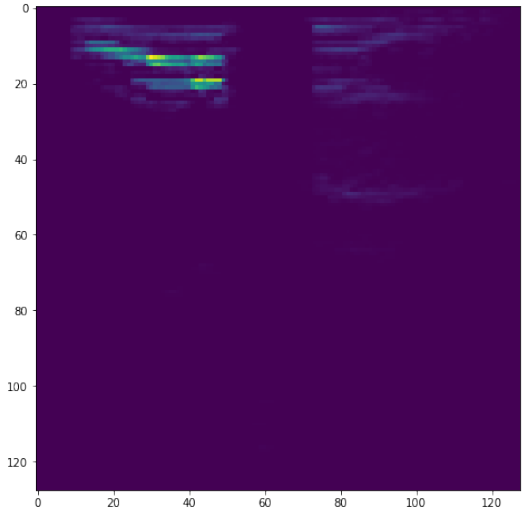

Figure 4: Transferred SG (Transferred Accent)

stylistically, which suggests that the accent has been transferred. We verified through synthesizing audio from the spectrogram which produces an accent audio similar to the intended target accent. However, due to lack of experience with the decoding process, we were unable to set the right set of parameters for the encoding and decoding algorithms which led to slightly garbled audio as output. Future work includes trying to find the right set of parameters to generate realistic speech.

As for accent classification, our discriminator model performs well on the test set. The t-SNE projection of test set examples can be found in Figure 6. This suggests our model is generally able to create a good decision boundary between American and British accents, and by extension can perform well as a discriminator in our end-to-end model. We were able to achieve over 95% classification accuracy on the test set.

After testing both WaveGlow and WaveNet, we proceeded to use WaveGlow in our pipeline because its inference was many times faster than WaveNet (1.7 seconds to generate 2 seconds of audio, compared to almost 10 minutes to generate 2 seconds of audio), and the reconstruction quality of the two methods was comparable. Experiments with WaveGlow demonstrated that it works well for reconstructing audio from mel-spectrograms derived directly from the audio. However, it produced only static when given mel-spectrograms produced by our generator. We determined this to be a

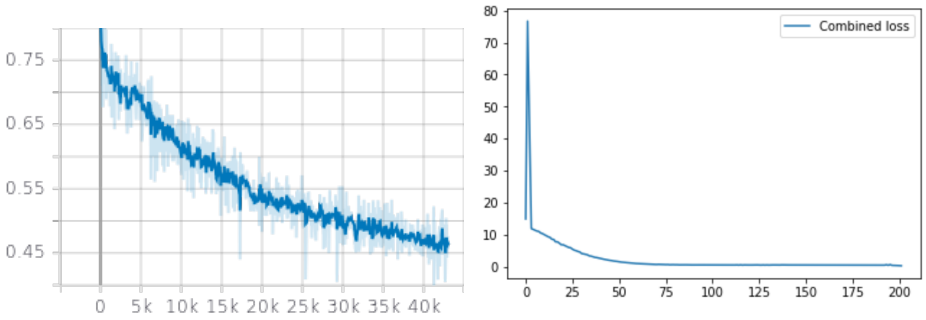

Figure 5: Training plots for the end to end model

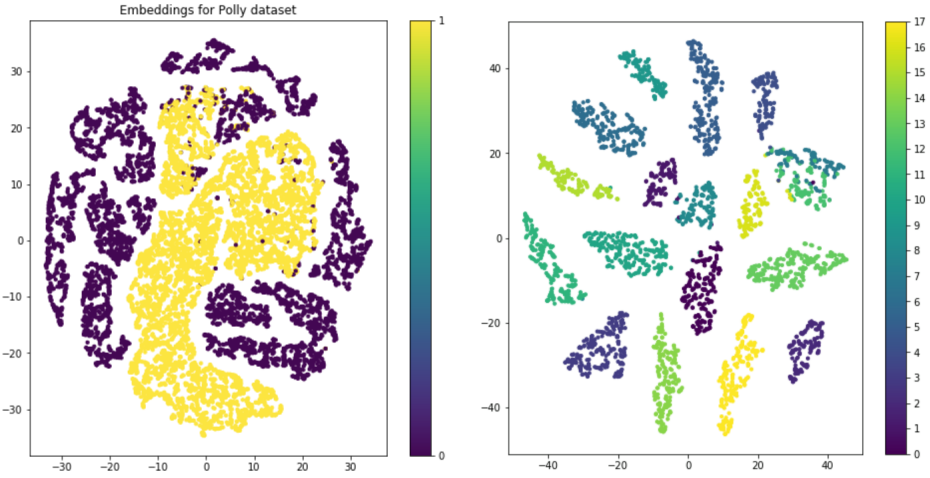

Figure 6: T-SNE projections of embeddings. On left, American (purple) and British (yellow) accents on the Polly dataset. On right, all accents in the CMU Arctic dataset.

product of the vocoder rather than the mel-spectrogram after trying the Griffin-Lim method to produce WAVs, which were decipherable to the human ear (although noisy, because Griffin-Lim is not as effective as deep methods).

Comparison of target audio passed through Griffin-Lim encoding and decoding with our generated audio, however, showed that our generated audio was extremely similar to the target audio, suggesting successful accent transfer. Spectrograms comparing these two quantities can be found in Figure 7, and audio samples can be found on our website at `http://carla.auton.cs.cmu.edu:5000/arch.html`.

6 Future Work

Although our results suggest that we were successfully able to perform accent transfer, listening to the audio will show that there is work left to be done. Furthermore, in the process of building this system, we thought of multiple extensions which we did not decide to pursue in the interest of time constraints. We now enumerate some of these which the interested reader may wish to pursue:

- Constrain mel-spectrogram parameters such that deep vocoders like WaveGlow, which are optimized for mel-spectrograms similar to those output by text-to-mel models like Tacotron2, can successfully produce audio.
- Achieve real-time or near-real time generation of transferred audio.
- Introduce support for multiple source and target accents.
- Allow for variable input sequence lengths rather than only single words.

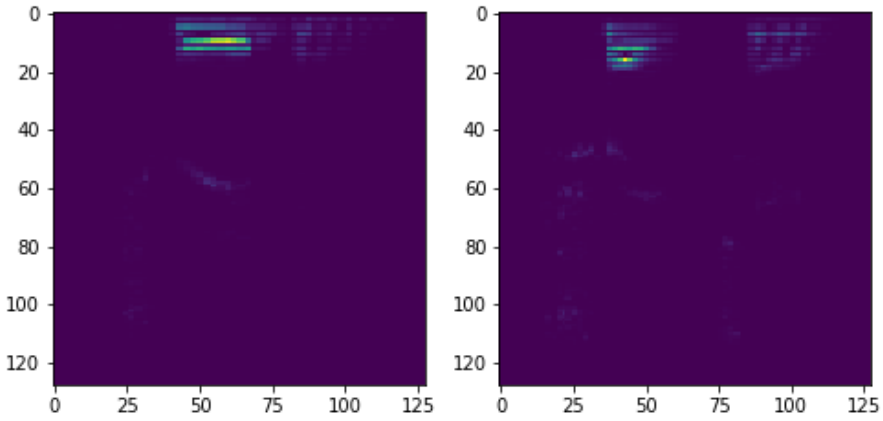

Figure 7: Sample spectrograms generated for the word *treaty* synthesized from Amazon Polly. **Left (source):** American accent, **Right (target):** British accent

References

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

Dongdong Chen, Lu Yuan, Jing Liao, Nenghai Yu, and Gang Hua. Stylebank: An explicit representation for neural image style transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1897–1906, 2017.

Fujun Luan, Sylvain Paris, Eli Shechtman, and Kavita Bala. Deep photo style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4990–4998, 2017.

Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. In *Advances in neural information processing systems*, pages 386–396, 2017.

Eric Grinstein, Ngoc QK Duong, Alexey Ozerov, and Patrick Pérez. Audio style transfer. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 586–590. IEEE, 2018.

Noam Mor, Lior Wolf, Adam Polyak, and Yaniv Taigman. A universal music translation network. *arXiv preprint arXiv:1805.07848*, 2018.

Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson. Autovc: Zero-shot voice style transfer with only autoencoder loss. In *International Conference on Machine Learning*, pages 5210–5219, 2019.

Yang Gao, Rita Singh, and Bhiksha Raj. Voice impersonation using generative adversarial networks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2506–2510. IEEE, 2018.

Sicong Huang, Qiyang Li, Cem Anil, Xuchan Bao, Sageev Oore, and Roger B Grosse. Timbretron: A wavenet (cyclegan (cqt (audio))) pipeline for musical timbre transfer. In *International Conference on Learning Representations*, 2019.

Ondřej Cífka, Umut Şimşekli, and Gaël Richard. Supervised symbolic music style translation using synthetic data. In *20th International Society for Music Information Retrieval Conference*, 2019.

Ye Jia, Yu Zhang, Ron J. Weiss, Quan Wang, Jonathan Shen, Fei Ren, Zhifeng Chen, Patrick Nguyen, Ruoming Pang, Ignacio Lopez Moreno, and Yonghui Wu. Transfer learning from speaker verification to multispeaker text-to-speech synthesis, 2018.

Abhinav Jain, Minali Upreti, and Preethi Jyothi. Improved accented speech recognition using accent embeddings and multi-task learning. In *Proc. Interspeech 2018*, pages 2454–2458, 2018. doi: 10.21437/Interspeech.2018-1864. URL `http://dx.doi.org/10.21437/Interspeech.2018-1864`.

Cong Zhou, Michael Horgan, Vivek Kumar, Cristina Vasco, and Dan Darcy. Voice conversion with conditional samplernn. *Proc. Interspeech 2018*, pages 1973–1977, 2018.

Hiromichi Kawanami, Yohei Iwami, Tomoki Toda, Hiroshi Saruwatari, and Kiyohiro Shikano. Gmm-based voice conversion applied to emotional speech synthesis. In *Eighth European Conference on Speech Communication and Technology*, 2003.

Kazuhiro Kobayashi, Tomoki Hayashi, Akira Tamamori, and Tomoki Toda. Statistical voice conversion with wavenet-based waveform generation. In *Interspeech*, pages 1138–1142, 2017.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

Amazon Polly. Amazon polly, 2020. URL `https://aws.amazon.com/polly/`.

Sajjad Abdoli, Patrick Cardinal, and Alessandro Lameiras Koerich. End-to-end environmental sound classification using a 1d convolutional neural network, 2019.

Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron J. Weiss, and Kevin Wilson. Cnn architectures for large-scale audio classification, 2016.

Daniel Griffin and Jae Lim. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(2):236–243, 1984.

Ryan Prenger, Rafael Valle, and Bryan Catanzaro. Waveglow: A flow-based generative network for speech synthesis. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3617–3621. IEEE, 2019.

Ce Qi and Fei Su. Contrastive-center loss for deep neural networks, 2017.