Style Transfer of Audio Effects

Mahya Khazaei
maahhi@uvic.ca

ABSTRACT

My project is based on this paper [1] presents a novel framework for style transfer of audio effects using differentiable signal processing. The framework uses a deep neural network to analyze an input recording and a style reference recording and predict the control parameters of audio effects used to render the output. This framework integrates audio effects as differentiable operators, performs backpropagation through audio effects, and optimizes end-to-end using an audio-domain loss. The proposed self-supervised training strategy enables automatic control of audio effects without the use of any labeled or paired training data. This framework has the potential to simplify the audio production process by allowing users to easily apply desired styles and effects to their recordings. In this project, after simple implementation of proposed model, I measured the capability of a filter-transformer which has been learned by limited types of filters.

1. INTRODUCTION

The potential applications of audio effects style transfer, this technique can be beneficial in various domains that rely on manipulating sound for creative or functional purposes. One such domain is the film and television industry, where audio effects are an integral part of creating immersive soundscapes that complement visual storytelling. By employing style transfer techniques, sound designers can quickly adapt the audio effects and production style from one scene or project to another, making it easier to establish consistent auditory aesthetics while saving time and resources.

Another application of audio effects style transfer is in the realm of music production. Artists and producers can use this technology to imbue their work with a unique sound signature, borrowing the style of one recording to inspire new creations. This could enable musicians to blend genres, experiment with different sound palettes, or re-interpret their own work with a fresh perspective. Additionally, podcast and audiobook production can also benefit from this technology, as it allows for seamless adaptation of audio quality and style between different episodes or chapters, ensuring a consistent listening experience for the audience.

In the realm of gaming, audio effects style transfer can enhance the gaming experience by allowing developers to create dynamic and adaptive soundscapes that respond to player actions and in-game events. By applying the style of specific audio samples to in-game sounds, developers can quickly generate cohesive and immersive audio environments that contribute to the overall game design and player immersion.

Furthermore, audio effects style transfer can be applied to the field of sound restoration and archiving. By transferring the style of well-preserved recordings to degraded or damaged ones, it is possible to recover some of the lost audio quality and ensure a consistent listening experience across a collection of historical or rare recordings. This technology could prove invaluable for preserving and revitalizing important cultural and historical sound artifacts, as well as enabling new artistic endeavors that draw from the past.

The process of audio production has long been an intricate and time-consuming task, requiring expertise in manipulating various audio effects such as loudness, timbre, spatialization, and dynamics. These effects, while powerful in the hands of experienced audio engineers, can be challenging for amateurs to navigate and often necessitate tedious adjustments even for professionals. Automatic audio production methods have emerged in response to these challenges, aiming to simplify and expedite the audio production process by providing adaptive control of audio effects based on input signals.

While rule-based systems built on audio engineering best-practices have seen success, they are limited by their inability to account for the vast diversity of real-world audio engineering tasks. Machine learning approaches offer greater flexibility but have been hindered by the difficulty of obtaining sufficient parametric data in a standardized way. Recently, deep learning has shown promise in overcoming these challenges, leading to an increasing interest in data-driven audio production techniques.

In the paper, DeepAFx-ST, a novel framework for audio effects style transfer is introduced that leverages differentiable signal processing to automatically control audio effects based on a short example style recording. This approach not only simplifies the audio production process but also generalizes to previously unseen recordings and varying sample rates.

The approach demonstrates the ability to perform audio effect style transfer for both speech and music signals, produce interpretable audio effects control parameters that facilitate user interaction, and operate at sampling rates different from those seen during training. As a person who wants to reimplement a simpler version of the whole project, it is beneficial to study the code, demonstration video, and listening examples provided online to facilitate further understanding and application in this domain.

2. BACKGROUND

Audio production style transfer has been an area of interest in recent research, with several studies focusing on controlling specific audio effects using techniques such as neural networks and random forest, as well as deep neural network approaches for controlling parametric frequency equalizers. However, these methods primarily concentrate on individual audio effects and use parameter domain losses, leading to certain limitations in performance and generalization across various effect classes.

Differentiable signal processing has emerged as a promising solution to overcome these challenges, allowing for an effective integration of digital signal processing (DSP) operations with neural networks. Parametric frequency equalizers (PEQ) and dynamic range compressors (DRC) are two common audio production effects that are of particular interest for differentiable audio effects. While differentiable PEQs have been previously developed, differentiable compressors remain unexplored. PEQs are typically designed as cascaded second order IIR filters, also known as biquads. However, the recursive filter structure may cause issues due to vanishing/exploding gradients and computational bottlenecks during backpropagation through time (BPTT), motivating the use of frequency-domain finite impulse response (FIR) approximations.

In addition to manual implementation of differentiable signal processing operations, alternative approaches such as neural proxy (NP), neural proxy hybrid models, and non-differentiable DSP implementations with numerical gradient approximation methods have been proposed. The NP approach trains a neural network to emulate the behavior of a signal processor, while hybrid models aim to reduce inference time complexity and minimize approximation error by combining neural proxy models with the original DSP device. Lastly, non-differentiable DSP implementations can be directly used with numerical gradient approximation methods, providing an alternative that does not require pre-training or knowledge of the DSP.

3. MODEL ARCHITECTURE

For someone looking to reimplement the production style transfer paper, the approach involves feeding magnitude spectrograms from input and style reference recordings into a shared-weight convolutional neural network encoder. This encoder generates a time series of embeddings for each recording, which are then aggregated using temporal average pooling to create a single embedding of dimension D for both the input and style reference. These embeddings are concatenated and passed to the controller network.

The controller network, a basic multi-layer perceptron (MLP), aims to produce control parameters that configure a set of audio effects, taking into account the information from the encoder about the production styles of the input and reference. The goal is to configure the audio effects such that the input signal, when passed through the effect chain, will produce a recording that matches the style reference.

A key aspect of this approach is integrating audio effects directly within the neural network's computation graph. This allows for incorporating domain knowledge, imposing a strong inductive bias, reducing processing artifacts, and lowering computational complexity. Unlike previous work, audio effects are fully integrated as differentiable operators or layers, enabling backpropagation through effects during training.

There are five unique differentiation strategies for backpropagating through audio effects to consider when reimplementing the paper: manually implemented automatic differentiation effects (AD), neural proxy effects (NP), full neural proxy hybrids (NP-FH), half neural proxy hybrids (NP-HH), and numerical gradient approximations (SPSA). While AD, NP, and SPSA methods have been used in automatic audio production tasks before, NP-FH has only been applied in static image processing hyperparameter optimization, and NP-HH is a novel approach. It's essential to compare these methods to determine their relative efficacy in the context of the reimplemented work.

4. IMPLEMENTATION

In this study, the project was implemented from scratch, beginning with the implementation of various parametric DSPs as filters, including Parametric Equalizer (PEQ), Compressor (CMP), Reverb, and Distortion. The PEQ was designed with six bands, ranging from 20 to 1200 Hz, and allowed the user to pass six different gains from -20 to 20 as the filter parameters. The compressor was simplified to accept only threshold (-20, 20) and ratio (2, 6) as input parameters, while the reverb accepted room scale from 0.1 to 1, wet level from 0.1 to 0.8, and dry level from 0.5 to 1.

To employ full neural proxy hybrids, a proxy was trained for each filter used within the architecture to mimic the filter and enable optimizer passage. CNN networks were utilized to train both the PEQ and CMP proxies. These proxies were integrated as a differentiable audio effect component within a pipeline. The component accepted input audio and eight parameters (six for the PEQ and two for the CMP), applying the PEQ proxy with the first six parameters on the input audio to generate the PEQ output. Subsequently, the CMP proxy applied compression based on the last two parameters to the PEQ output, producing the final output.

The model architecture employed CNN as encoders and MLP for the controller. The majority of neural network layers in this project utilized ReLU activation functions, while the controller's final output, which determined the parameter, employed a sigmoid and scaler to facilitate training by providing parameter range hints.

A subset of the train-clean-360 dataset, which is part of LibriTTS, was used for this project. It was divided into 1,000 audio tracks with a length of 0.53 seconds and a sample rate of 24,000. The time domain loss was computed using the mean absolute error (MAE), while the frequency domain loss was calculated using the multi-resolution short-time Fourier transform loss (MR-STFT). The MR-STFT loss is the sum of the distances between the STFT of the ground truth and estimated waveforms, measured in both log and linear domains across multiple resolutions, with window sizes $W \in [32, 128, 512, 2048]$ and hop sizes $H = 256$.

5. RESULTS AND EVALUATION

To evaluate the performance of the implemented DSPs, they were applied to multiple audio tracks, and the parameters were manually adjusted to observe if the resulting audio changed accordingly. Due to the absence of data augmentation and the relatively small dataset, there was a high risk of overfitting. Training epochs were halted when the loss of the validation dataset began to increase.

Upon training the main network with proxies, the performance was validated against DSP functions using three distinct approaches: (1) generating audio tracks similar to their corresponding PEQ and CMP processed counterparts, (2) generating audio tracks resembling their reverb-processed counterparts, and (3) generating audio tracks similar to their distortion-processed counterparts. The resulting MAE and MR-STFT for applying PEQ and CMP were 0.0539 and 0.2703, respectively. For the reverb approach, the values were 0.0686 and 0.4552, while for distortion, they were 10798.12 and 79496.91.

Upon listening to the results, it was evident that a style transfer model with an equalizer and compressor could not successfully apply other filters, such as reverb and distortion. Although the numerical results for the reverb approach were acceptable, the generated audio did not closely resemble the reference. This discrepancy can be attributed to the loss functions' inherent characteristics, which aim to measure similarities in time and frequency domains. Distortion introduces significant noise with varying frequencies and alters the audio in the time domain, thereby affecting the loss functions' ability to provide a meaningful comparison.

6. REFERENCES

[1] Steinmetz, Christian J., Nicholas J. Bryan, and Joshua D. Reiss. "Style transfer of audio effects with differentiable signal processing." arXiv preprint arXiv:2207.08759 (2022).

[2] T. Wilmering, D. Moffat, A. Milo, M. B. Sandler, “A history of audio effects,” Applied Sciences, vol. 10, no. 3, p. 791 (2020).

[3] M. M. Ram´ırez, E. Benetos, J. D. Reiss, “Deep learning for black-box modeling of audio effects,” Applied Sciences, vol. 10, no. 2, p. 638 (2020).