ChatGPT

Prior Work on Learning Audio Transformations from Paired Examples

Neural Tape Recorder Emulation

One closely related work is the **neural emulation of analog tape machines**. Mikkonen *et al.* (2023) introduced a deep learning approach to replicate the sound of magnetic tape recorders [1]. Their system models the key tape characteristics – **nonlinear tape saturation**, **"wow/flutter" time-varying delay**, and **tape hiss noise** – entirely from data. In practice, they used a recurrent neural network to learn the tape's **hysteretic distortion** behavior, while separate U-Net convolutional networks handled the **fluctuating delay and noise components** [1]. This hybrid model, trained on paired "clean vs. tape-recorded" audio, was shown to **faithfully capture the vintage tape recorder's character** according to objective metrics [2]. The result is a convincing digital **"tape style" transfer** that can serve as a *virtual analog* of a reel-to-reel machine [2]. (Notably, Chowdhury's *ChowTape* project took a related approach: it integrates a small **State Transition Network** to emulate the tape's magnetic hysteresis curve in real-time, demonstrating another use of neural nets for tape distortion within a plugin [3] [4].) Overall, these works confirm that a neural model can learn the **Revox A77's** warmth and saturation from nothing more than paired input-output recordings of clean vs. tape-processed audio.

Black-Box Neural Modeling of Audio Effects (Music Domain)

Beyond tape machines, there is a rich history of using neural networks to **learn audio effect transformations** from example pairs. In the music domain, researchers have successfully modeled **guitar tube amplifiers, distortion pedals, and other analog effects** by treating them as black boxes – feeding a clean audio input into the real device and training a network to reproduce the processed output. For example, Wright *et al.* (2020) compared a WaveNet-style convolutional model against an LSTM recurrent model for guitar distortion **amp modeling**, and found that both could achieve excellent fidelity [5]. Remarkably, they showed that **as little as three minutes of audio data** was sufficient to train these models to emulate a given device's sound [6]. The resulting neural nets were so accurate that in listening tests many users **could not distinguish** the model's output from the real amplifier, and they ran in real-time on a normal PC [7]. This demonstrates that **high-quality audio style transformation is achievable on a single computer**, without needing massive datasets or clusters. In fact, Wright *et al.* (2019) earlier showed that even a very compact model – a single-layer **recurrent neural network** with one fully-connected output layer – could emulate nonlinear guitar effects **with accuracy matching a WaveNet**, but at a fraction of the computational cost [8]. This RNN-based approach was explicitly designed for **real-time audio**, proving that efficient models in PyTorch/TensorFlow can learn an effect's input→output mapping and run with low latency on standard hardware [8]. Subsequent studies have refined these architectures (e.g. dilated **Temporal Convolutional Networks** and deeper LSTMs) but the theme remains: **train on paired dry/wet audio and directly learn the effect's transformation**. Most work converges on either **WaveNet-convolutional networks or RNN (LSTM/GRU) networks**, both of which have shown **sufficient accuracy and low latency for real-time inference on consumer hardware** [9]. An illustrative open-source project is the *Neural Amp Modeler (NAM)*, which lets users capture their own amplifier's sound by training either a WaveNet-style or LSTM model on input/output recordings [9]. NAM and similar tools demonstrate that common ML libraries (PyTorch) can be used to train these models on a single GPU, and then deploy them as audio

plugins. This body of work indicates that your approach – training a model on paired clean and "Revox A77" audio – stands on firm ground. Researchers have **successfully transferred analog "style" onto digital audio** for instruments (guitar, drums, etc.) using supervised learning, and achieved perceptually transparent results [7].

Audio Style Transfer Techniques (Music and Speech)

In a broader sense, the task of transforming audio to adopt a certain *style* or sonic character has been explored in both music and speech contexts. One line of research directly targets **audio production style transfer**: for instance, Steinmetz *et al.* (2022) present a framework to impose the **effects chain and tonal coloration** from one recording onto another [10]. Their system uses a deep network to analyze an input signal and a "style reference" signal, then predicts the settings of differentiable audio effect modules (EQ, compression, saturation, etc.) to make the input sound like the reference [10]. Notably, their approach is *self-supervised* (no one-to-one paired samples required); by integrating differentiable DSP effects into the model, they train end-to-end on unlabeled audio, directly minimizing an audio-domain loss. They demonstrate convincing **production style transfer on both speech and music**: the trained model generalizes to unseen audio and even new sample rates, successfully making a clean voice recording sound "produced" (with the timbre and loudness profile of a polished reference) [11]. This shows the feasibility of style transfer in the **speech domain** as well – e.g. making a modern vocal recording take on a *vintage* or analog character – using neural networks. There have also been earlier proofs-of-concept for **audio *style transfer*** inspired by image style transfer. Grinstein *et al.* (2018) treated style transfer as an optimization problem: they extracted summary statistics (a "texture model") from a reference audio and then **iteratively synthesized a new signal** that injects those texture characteristics into a target sound [12]. In their case no neural network was trained; instead, they used a sound texture loss (on spectro-temporal features) to transform, say, a speech recording to adopt the texture of a noisy reference. While such optimization-based methods confirmed the *concept* of audio style transfer [12], modern approaches like those above focus on **training a neural model** so that the transformation can be done quickly and learn more complex styles.

In summary, **prior work strongly supports the viability of learning an audio transformation from paired input-output data**. Researchers have used supervised deep learning to capture the subtle nonlinearities of analog tape machines [1] [2], vacuum-tube amplifiers and distortion pedals [6] [7], and even entire production chains. These models – often based on convolutional or recurrent architectures – can be trained on a single machine with standard Python ML frameworks, given a reasonable collection of example pairs. By leveraging these techniques, your project can perform **audio style transfer** (for both music and speech content) to convincingly emulate the warm, vintage sound of a Revox A77 tape recorder. The literature indicates that with a well-designed network and sufficient paired examples, the model can learn the tape recorder's sonic fingerprint and apply it to new audio, all within the compute limits of a personal workstation [13] [8].

Sources:

- Mikkonen, O. *et al.* (2023). *Neural modeling of magnetic tape recorders* – proposed an RNN+U-Net architecture to emulate reel-to-reel tape (nonlinear saturation, wow/flutter, hiss) from input-output audio data [1] [2].
- Wright, A. *et al.* (2020). *Real-Time Guitar Amplifier Emulation with Deep Learning* – demonstrated WaveNet and LSTM models that learn guitar amp/pedal transformations from only minutes of audio, achieving perceptually transparent, real-time results on a PC [6] [7].

- Wright, A. *et al.* (2019). *Black-Box Modeling with Recurrent Neural Networks* – showed a simple recurrent network can mimic nonlinear analog effects (e.g. Big Muff pedal) with WaveNet-level accuracy but lower CPU usage, suitable for real-time inference [8].
- Steinmetz, C. *et al.* (2022). *Style Transfer of Audio Effects with Differentiable Signal Processing* – introduced a model to apply the production "style" of a reference audio onto a target, by predicting effect parameters; achieved convincing style transfer on both music and speech audio without direct pairwise training data [10] [11].
- Grinstein, E. *et al.* (2018). *Audio Style Transfer* – early work formulating audio style transfer as an optimization problem using texture statistics of a reference audio to impart its sound characteristics to a target signal [12].
- **Additional**: Chowdhury, J. (2020). "*Tape Emulation with Neural Networks*" – blog describing the ChowTape plugin, which uses a learned *state-transition network* to model tape hysteresis efficiently [3] [4]. Also, various open-source projects (e.g. Neural Amp Modeler) demonstrate practical training of WaveNet or LSTM effect models on everyday hardware [9].

---

[1] [2] Neural modeling of magnetic tape recorders
https://www.dafx.de/paper-archive/2023/DAFx23_paper_51.pdf

[3] [4] Tape Emulation with Neural Networks | by Jatin Chowdhury | The Startup | Medium
https://medium.com/swlh/tape-emulation-with-neural-networks-699bb42b9394

[5] [6] [7] [13] Real-Time Guitar Amplifier Emulation with Deep Learning | MDPI
https://www.mdpi.com/2076-3417/10/3/766

[8] Real-Time Black-Box Modelling With Recurrent Neural Networks
https://dafx.de/paper-archive/2019/DAFx2019_paper_43.pdf

[9] Parametric Neural Amp Modeling with Active Learning
https://arxiv.org/html/2509.26564v1

[10] [11] [2207.08759] Style Transfer of Audio Effects with Differentiable Signal Processing
https://arxiv.org/abs/2207.08759

[12] [1710.11385] Audio style transfer
https://arxiv.org/abs/1710.11385