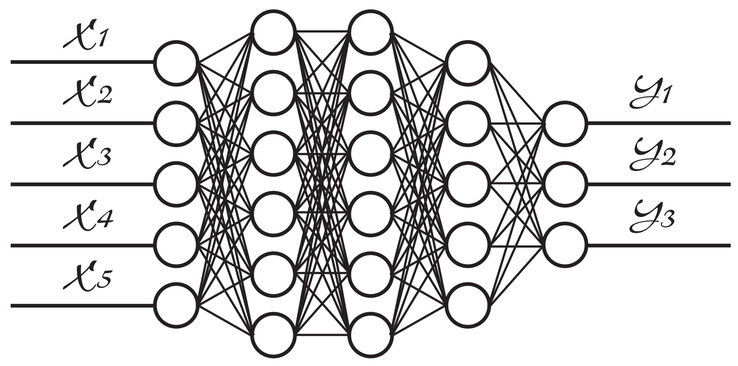
Convention Paper #9846

Anne

Improving Neural Net Autoencoders for Music Synthesis

What is this?

ANNe is an audio plugin that uses an Autoencoding Neural Network as an audio effect. Users can alter the parameters of the neural network to create new, unexplored sound domains.

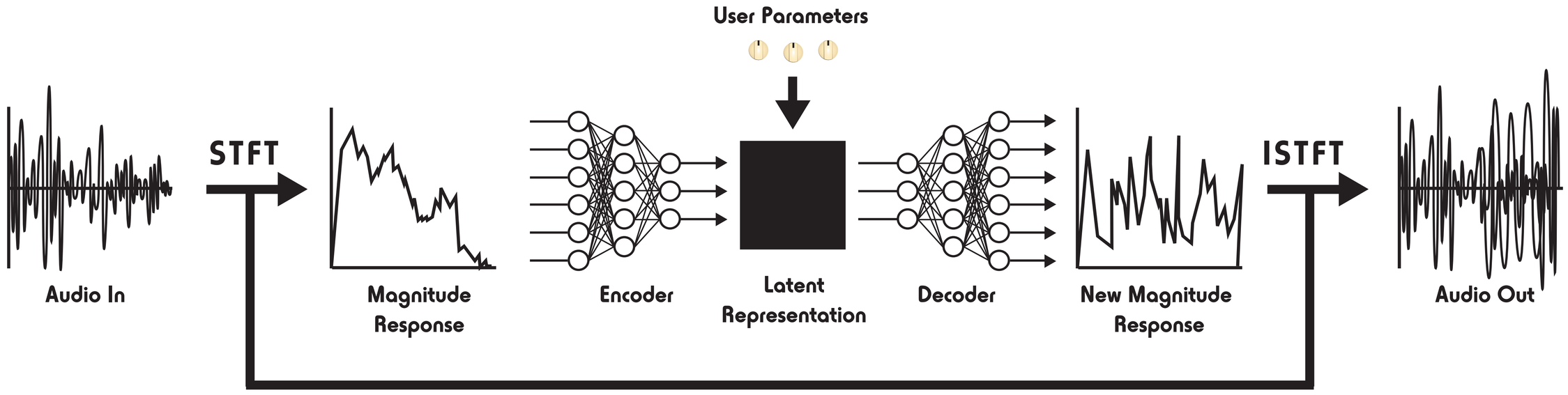


Neural Network

How does that work?

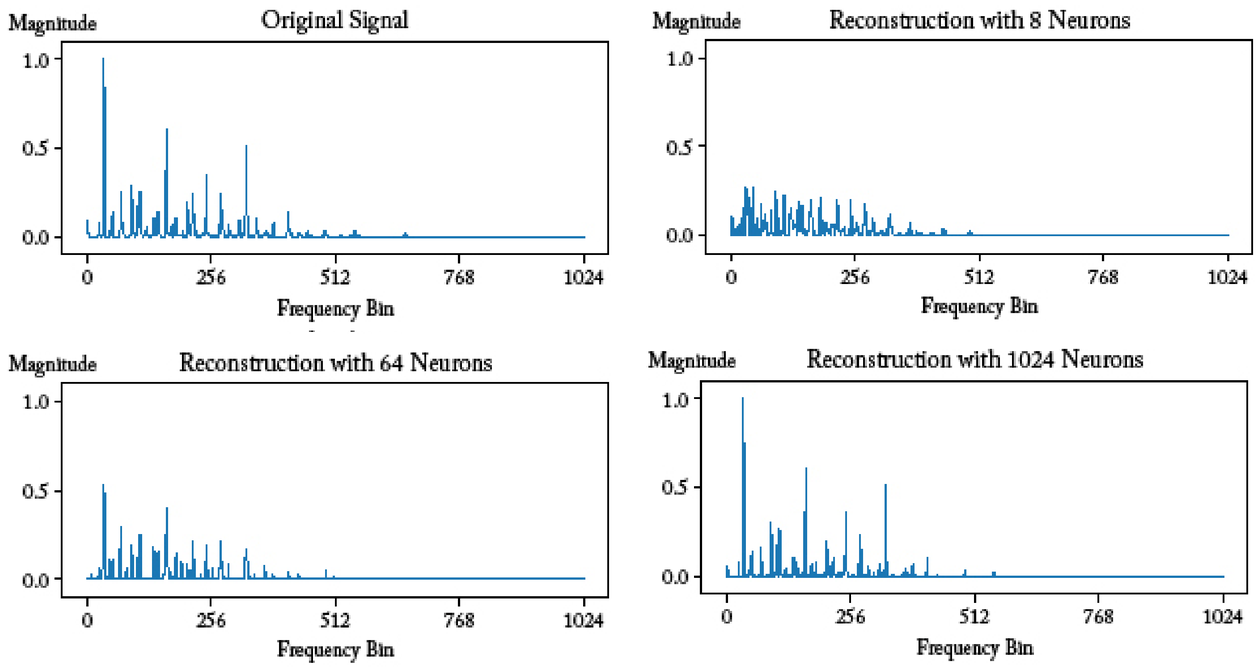
The work presented here is based off Andy Sarroff's paper "Musical Audio Synthesis Using Autoencoding Neural Nets"

The autoencoding neural network is designed to encode an input audio signal as a high level representation it can later decode. Throughout training, the neural network creates a "latent space" that contains representations of musical signals. By modifying the parameters of the neural network, a user can shift an input's high level representation within the latent space, thus creating new sounds when decoded.



How was the neural net designed?

First we created a training corpus of 70,000 normalized magnitude Short Time Fourier Transform (STFT) frames using audio generated from a microKORG synthesizer. All audio processing was handled using the LibROSA library in python. Then, we designed an autoencoding neural network topology in TensorFlow – the neural network is trained to encode and decode these STFT frames in a lossless manner. While designing, we combined Sigmoid and Rectified Linear activation functions to improve the neural net's performance. Ultimately we engineered a reasonable tradeoff between the re-constructive capabilities of the neural net and the robustness of the latent space it generates.



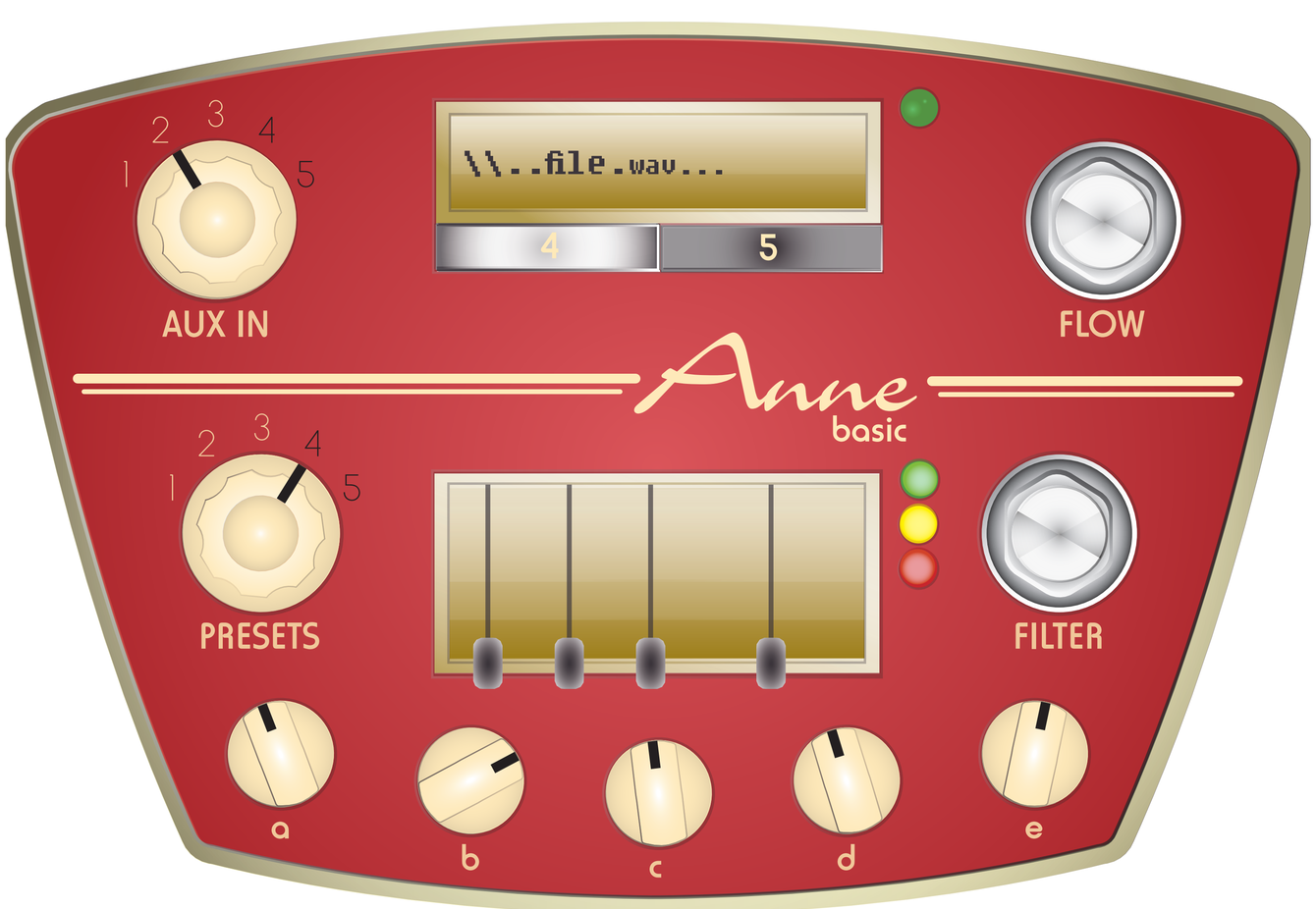
Citations

A. Sarroff, "Musical audio synthesis using autoencoding neural nets," Dec 2015. [Online]. Available: http://www.cs.dartmouth.edu/~sarroff/projects/autoencoding-synthesizers/

M. Abadi, "Tensorflow: Learning functions at scale," ICFP, 2016. [Online]. Available:http://dl.acm.org/citation.cfm?id=2976746

B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto,"librosa: Audio and music signal analysis in python," in Proceedings of the 14th python in science conference, 2015, pp. 18–25.

How to interface with the neural net?



In designing an interface, we wanted to allow users to experience the full potential of the neural network without burdening them with a learning curve. Therefore, we drew inspiration from guitar pedals to inform the GUI design.

ANNe comes packaged with 3 default input signals – A piano middle C, a snare drum, and a guitar chord. Users can choose between these 3 inputs, or their own sounds on options 4 or 5, as their input signal.

Also, ANNe comes with preset topologies 1-3. These presets alter how the input gains a-e are sent to the neural network. Both the gains and the topology alter the sound of the output signal. Each preset was found through much testing and has its own aural footprint. Thus users can design sounds by choosing an input signal, a preset topology, and tuning the knobs a-e.

Should advanced users want more control, they can alter how parameters are sent to the neural network by dragging the sliders in the center of the pedal. These users can save their topology design as presets 4 or 5.

Pressing "flow" will play a bypassed version of the signal, and pressing "filter" will play the output of the neural network.

Joseph Colonel Benjamin Sterling Sam Keene Chris Curro Yonatan Katzelnik