Notes on references

Juan Corredor

October 21st, 2024

1 DDSP

[1] used as main reference for the research.

1.1 Important bullet points

- Interpretable and modular approach to generative modeling

- Classic signal processing elements with deep learning methods

- Models that rely on strided convolution or windowing (STFT) need to align wave-shapes or suffer from spectral leakage.

- DDSP takes the approach of vocoders in using oscillators to synthesize signals.

- According to the DDSP paper [1], the DDSP library is capable of extrapolating timbres not seen during training, and independent control over pitch and loudness during synthesis.

- References to Neural Source Filter [2] imply this might be a good additional reference for research.

1.2 Newly learned concepts from this document

- Strided Convolution:

  Convolution that has a hop length, meaning, it skips some information to avoid analyzing redundancies.

- Teacher forcing:

  Feeding back the correct answers into training algorithms to reduce training times and lead the model in the right direction during training.

- Automatic differentiation:

  Also: 'algorithmic differentiation' or 'computational differentiation' means splitting derivatives into simpler operations using the chain rule.

- Deterministic autoencoder

- Adversarial training

- Variational inference

- Jacobian design

- Stochastic latents

- CREPE model

- Latent encoding

1.3 How it relates to my research

I believe the general structure and inner workings of the proposed DDSP synthesizer from the paper are exactly how I need to model my own synthesizer.

The proposed system requires an audio database to train the machine learning model and then use that model to extract the required features to build a faithful additive synthesizer. What's interesting about this system is that it allows for the creation of timbres beyond the training set, meaning that theoretically, if my training database consists of a collection of plucked string samples, I could interpolate timbres between them or exaggerate some of their features beyond what I have already recorded.

Since this model has been tested with longer audio samples, so far a lot of the development has been using audio sampled at relatively low sample rates (in the range of the 16kHz sample rate). However, due to the narrow scope of my project, I believe I should be able to use samples recorded at higher sample rates and generate high-quality output.

Another interesting aspect of the DDSP library is that it allows for flexible scalability. Meaning, I could generate consistent output with very few parameters and relatively low computational power.

References

[1] J. Engel, L. H. Hantrakul, C. Gu, and A. Roberts, "Ddsp: Differentiable digital signal processing," in *International Conference on Learning Representations*, 2020.

[2] X. Wang, S. Takaki, and J. Yamagishi, "Neural source-filter-based waveform model for statistical parametric speech synthesis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5916–5920, IEEE, 2019.