A COMPARATIVE EVALUATION OF TONAL REPLICATION TECHNIQUES FOR MUSIC COMPOSITION AND SOUND DESIGN THROUGH TONAL SYNTHESIS

A Thesis

Presented to the

Department of Information Systems

and Computer Science

Ateneo de Manila University

In Partial Fulfillment

of the Requirements for the Degree

Bachelor of Science in Computer Science

by

Angelo Joaquin B. Alvarez

John Aidan Vincent M. Ng

Justin Carlo J. Reyes

2024

ABSTRACT

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

CHAPTER 1

Introduction

1.1 Context of the Study

Music is an art that evolves as musicians search for new ways to express their messages, emotions, or even their identity. Throughout music history, there has been a consistent push towards innovation whether it be by composition, sound design, or other areas of study. One such example is the musical expression of jazz, considering its relatively recent rise in the early 1900's. In the past half century, synthesizers have led to the generation of audio signals and waveforms using additive, subtractive, or frequency modulation synthesis techniques.

The advent of synthesizers gave rise to a new era of music-making, giving birth to an array of new genres such as electronic, techno, ambient, and house music, among others. Oscillators, filters, envelopes and modulation sources came with the rise of synthesizers to empower musicians in sculpting sounds to the limits of their creative expression. Through the advancements of these features, synthesizers have broken the barriers of genres. Synthesizers have been integrated into various musical contexts, giving each genre the modern [smth].

In the past decades, the dominance of synthesizers in the music industry cannot be understated. Nevertheless, new methods of music production have come to light due to advancements in synthesizer technology and artificial intelligence. Methods such as algorithmic composition, modular synthesis, sample-based production and Machine Learning and Artificial Intelligence-assisted Composition have increased in usage over the past years.

This study aims to explore the intersection of artificial intelligence and synthesizer technology, focusing on the replication of instrumental tones through an audio synthesis from an instrumental melody. Through comparing the latest methods and techniques in the field of tone replication, this study aims to contribute to our understanding of how synthesizers can further shape the landscape of music composition and sound-design in the coming years.

1.2 Research Objectives

The study aims to utilize various techniques in audio synthesis and replication, namely Tone Transfer, GANs (Generative Adversarial Networks), and DSPGAN in order to replicate instrumental tones, comparing them in terms of accuracy and efficiency. The sub-objectives are as follows:

- ❖ To determine how the quality and choice of input audio encoding and representation affect sound replication efficiency and accuracy,
- ❖ To determine the stability of the outlined methods in handling various factors in the audio input, such as noise, frequency masks, low audio quality, and fragmented audio inputs,
- ❖ To determine the perceptual quality of synthesized audio as compared to the original audio input, and
- ❖ To determine the effect of contrasting feature selection and representation on the accuracy and efficiency of synthesized audio.

1.3 Research Questions

The study seeks to answer the question: How do methods in audio synthesis and replication such as Tone Transfer, GANs, and DSPGAN compare in their accuracy and efficiency in replicating instrumental tones? In answering this question, the following sub-questions can be answered:

- ❖ How does the quality and the choice of encoding and representation for the input audio impact the efficiency and accuracy of sound replication?
- ❖ How stable are different methods in audio synthesis and replication in handling noise and frequency masks in the audio input, low audio quality, and fragmented audio inputs?
- ❖ How does the synthesized output generated by the model compare to the original audio input in terms of perceptual quality?
- ❖ How do the contrasting feature selection and representation employed in different methods contribute to their accuracy and efficiency in replicating instrumental tones?

1.4 Scope and Limitations

This study will be limited to implementations of the concerned methods (Tone Transfer, GANs, and DSPGAN) as outlined in their respective studies.

I will not Paragraph Muna:

- This study is focused on the specific implementations of:
    - Tone transfer architecture as delineated in [Synthesis] FM Tone Transfer with Envelope Learning.pdf (Caspe et. al, 2023)

    - A Generative Adversarial Network architecture as specified in [Synthesis] GanSynth - Adversarial Neural Audio Synthesis.pdf (Engel et. al, 2019)
    - Variational Autoencoders architecture as specified in [MAY STUDY BA].
- The Audio Input is further specified as:
    - Any melodic recording of an instrument.
    - TENTATIVE Instrumental recordings will use the following instruments:
        - Violin
        - Guitar
        - Flute
        - Trombone
        - Organ
        - Marimba
    - Quality of audio varies to determine the stability of each architecture.
    - Note Qualities also vary to determine which features of each architecture affect the audio output
- The dataset to be used will be the NSynth Dataset consisting of 305,979 musical notes with unique pitch, timbre and envelope played from 1,006 instruments.
- Metrics to be used to determine the accuracy of the output from the input are:
    - Human Evaluation (tentative)
    - Frechet Inception Distance (FID)
    - Number of Statistically-Different Bins (NDB)
- Limitations include
    - There is no single metric that could effectively determine the accuracy of the models. As such these scores may not capture all aspects of accuracy within tonal replication.
        - Human evaluation to determine output quality can be subjective and produce unexpected or varied results
        - Inception Score does not capture perceptual accuracy
        - Pitch Metrics ignores timbre and nuanced musical characteristics
        - NDB ignores temporal aspects and focuses on spectral content
    - Results may be overfit to the NSynth Dataset (tbd by the models itself)

        - Unsure cause replication nga dba so dat magoverfit talaga 😕
    - This study focuses on the comparison of specific implementations of architecture in each neural network and does not delve into deeper and more advanced techniques such as unique model and architecture creation.

1.5 Significance of Study

In the intersection of computer science and the area of computational musicology, this study aims to further the field of music sound synthesis using machine learning by figuring out how

CHAPTER 2

Review of Related Literature

Important?

- https://research.facebook.com/publications/sing-symbol-to-instrument-neural-generator/
- https://dl.acm.org/doi/pdf/10.1145/3616195.3616196
- https://openreview.net/attachment?id=B1x1ma4tDr&name=original_pdf
- https://openreview.net/pdf?id=H1xQVn09FX ← Metrics

2.1 Previous Methods for Audio Synthesis

2.1.1 WaveNet and GANs (Generative Adversarial Networks)

The use of WaveNet autoencoders for neural audio synthesis, one of the recent technological advancements regarding audio synthesis of the past decade, paved the way for computational musicology by using conditional autoencoders learned from raw audio waveforms. One of its contributions is also the NSynth dataset, a "large-scale dataset for exploring neural audio synthesis of musical notes." which was composed of over 300,000 notes belonging to instruments of different families (strings, vocals, wind instruments, etc.). It was found in their study that playing styles such as vibrato could be replicated looking at instantaneous frequencies in a spectrogram, and harmonic structures and overtones blended more smoothly. While revolutionary, it was only able to recreate sample-based audio, and was not able to capture fully global context. Engel et al. (2017)'s paper showed that creating sample-by-sample audio signals was possible through the use of deep learning.

Instead of using vocoders, frequency modulation, MIDI synthesizers, or any combination of the three and other possible methods, deep learning for audio synthesis is headed toward directly replicating the waveform of audio samples. GANSynth, a study conducted by Engel et al. (2019) uses a Progressive GAN architecture combined with conditioning of an additional feature: a one-hot representation of musical pitch. GANSynth uses Short Time Fourier Transforms (STFT)

and IF-Mel (log magnitudes and mel frequency scales) variants in order to generate samples that are over 50,000x faster than those generated by WaveNet. Aside from faster sample-generation, GANSynth utilizes information from the latent features and musical pitch of the training dataset to generate audio exhibiting smooth timbral interpolation and timbral consistency across different pitches. The introduction of GANSynth marks a significant development in the use of GANs for audio generation, synthesizing audio with superior quality when compared to the previous WaveNet autoencoder.

2.1.2 DDSP (Differentiable Digital Signal Processing)

In 2019, the term differentiable digital signal processing (DDSP) was introduced by Engel et al. (2020). In their study, the pitfalls of various methods were discussed, like strided convolution models (WaveGAN, SING) giving only general representations that can present any waveform, fourier-based models (GANSynth) not being able to assert the issue of spectral leakage leading to phase misalignments, and autoregressive models (WaveNet, RNNs) being able to hold its' own but being prone to exposure bias and incompatibility with other spectral audio features.

The DDSP library introduced by Engel et al. offers a suite of differentiable components, including Spectral Modeling Synthesis, Harmonic Oscillators, overlapping Hamming window-based amplitude envelopes, a time-varying FIR filter, a subtractive synthesizer, and a reverb module. Evaluated with both supervised (NSynth) and unsupervised (solo violin) datasets, DDSP exhibits high-fidelity synthesis capabilities. Crucially, it enables independent control of loudness and pitch while facilitating timbral transfer (e.g., transforming a singing voice into violin-like sounds). The entire architecture of the autoencoder essentially channels multiple audio signals through different audio synthesis algorithms and combines each signal into one using a reverberation module.

In recent years, DDSP has been retrofitted into various methodologies in the academic field, as discussed by Hayes et al. (2023)'s catalogue and survey into the use of said techniques in sound and music synthesis. According to their review, DDSP-based audio synthesis in the field of musical audio synthesis can be classified into three distinct areas: (i) Musical Instrument Synthesis, (ii) Performance Rendering, (iii) Timbre Transfer, and (iv) Sound Matching. For example, Hayes et al. (2021)'s neural waveshaping synthesis method uses NEWTs (neural waveshaping unit, learning shaping functions from unlabelled audio) fed into a harmonic-plus-noise synthesizer, much like DDSP. Another paper created a real time implementation of the library using PyTorch, Masuta and Saito (2021) implemented a similar-yet-controllable version by having controllable parameters such as cutoff frequency and a differentiable subtractive synthesizer, and more. Additionally, Shan et al. used DDSP methods as a building block, combining it with differentiable wavetable synthesis (essentially ???

DSPGAN, a GAN-based universal vocoder for high-fidelity speech synthesis by applying the time-frequency domain supervision from digital signal processing (DSP), is one of the most recent developments combining both GANs and methods from Engel's methodology. The paper discussed how pitch jitters and discontinuous harmonics would commonly be found in GAN-based vocoders, and so they combined a DSP module, finding that " it can generate high-fidelity speech for various TTS models trained using diverse data."

[TONE TRANSFER IF GAGAWIN PA]

2.2 Audio Synthesis Evaluation Metrics

Despite the development of generative models in audio, video, and image synthesis, finding evaluation methods still pose a challenge, particularly for implicit models that do not generate quantifiable values. Although these models can easily be judged through perceptual evaluation,

objective metrics are still significant in the comparison of models, architectures, and hyperparameters. Common evaluation metrics of generative models are often driven by intuition and may overlook certain features of the product. These limitations are also present in audio synthesis. Vinay and Lerch cite various metrics that can be used for evaluating audio generative models following methodologies of previous studies. Moreover, Betzalel et al. recommends using a diverse set of evaluation metrics when comparing generative models to control the variability in scores. As such, not only objective evaluations are necessary, subjective evaluations, such as those requiring human discrimination, are also necessary for ensuring that the produced audio will cater to the perception of humans.

2.2.1 Objective Evaluation Metrics

Numerous metrics have been developed to evaluate generative models, with Inception Score (IS) being the most commonly used. Betzalel et al. delineates six (6) evaluation metrics used for implicit generative models. The paper further recommends dropping the use of IS for Frechet's Inception Distance (FID) as IS performs relatively worse among other evaluation metrics. However, this study is primarily focused on image generative models. IS, FID, and other metrics that utilize Inception measure the quality and diversity of an image. Using metrics such as IS and FID for other generative models, such as audio and video, requires replacing the Inception component of these metrics with features of the intended input data.

However, despite being originally developed for image evaluation, these metrics have been modified to adapt to audio-based generative models. A workaround to adapt FID to the audio domain has been developed by Kilgour et al. by replacing the inherent Inception network within FID with the VGGish model. The developed technique, called Frechet Audio Distance (FAD), works by generating embedding statistics with the VGGish model on the evaluation set and

compares it to the embedding statistics generated on a reference set of clean music, which is usually the training set. This metric is used to measure audio quality by analyzing distortions within the audio. Thus, this metric will be helpful in objectively evaluating the quality of produced audio from the three generative models.

The Frechet Audio Distance is not sufficient to fully evaluate the chosen generative models, as it only compares the distribution of the generated audio to the distribution of the original audio dataset. Another metric called Number of Statistically Different Bins (NDB/*k*) that was introduced by Richardson and Weiss to measure the diversity of generated audio from the original dataset. NDB/*k* works by distributing test samples into *k* clusters using an *L2* distance measure and performing two-sample *t*-tests between each cluster pair, with the NDB score being the proportion of statistically different clusters to the amount of clusters . This metric measures that the produced audio is uniquely generated and not replicated from the original dataset.

The use of these objective evaluations allows for a nuanced comparison between generative models. While FAD provides valuable insight into audio quality by measuring distortion from the training data, NDB/*k* offers a complementary perspective by assessing the diversity of generated audio, measuring its similarity to the training data. By considering both metrics, we gain a more comprehensive understanding of an audio generative model's strengths and weaknesses.

2.2.2 Human Evaluation Metrics

Since the generated audio will be made for consumption and use for music creation, human evaluation is necessary to ensure accurate perceptual quality. A commonly used subjective metric is the Mean Opinion Score (MOS) where participants are asked to rate the sound they hear in a 1 to 5 Likert Scale across a set of questions that pertain to the quality of the audio. These questions might focus on aspects like clarity, naturalness, pleasantness, and overall fidelity. By aggregating the scores

from a diverse group of listeners, MOS provides a quantitative measure of how humans perceive the generated audio. Combining these objective and subjective evaluations provides a well-rounded picture of the strengths and weaknesses of different audio generative models.

[OTHER PART OF TESTING]

CHAPTER 3

Methodology

https://lucid.app/lucidchart/39cf1906-a300-44a6-b6e8-2be9975e22a6/edit?viewport_loc=-11%2C-11%2C2032%2C1289%2C0_0&invitationId=inv_03d5bddd-6d99-4076-8577-79b10c15d05d

- Nsynth dataset
- Envelope Learning:
    - Instead of using oscillators from generated patches, we could use:
        - Chroma Features?
    - Model Inference (3.3)
        - Feature extraction....
            - **WHERE THE FUCK IS THIS IN THE CODE??????**

(2) Control Prediction, we use our neural network g_ϕ to infer a set of frame-wise FM synthesis controls, the oscillator output levels $\hat{ol}_k$, from the conditioning signals $\hat{a}_k$ and $\hat{f}_k$.

$$\hat{ol}_k = g_\phi(\hat{a}_k, \hat{f}_k) \quad (3)$$

-
            - This is from Tone Transfer with Envelope Learning: however, it uses **generated synth patches** while we are working with raw data...
            - Possible alternative:
                - a^k <- loudness
                    - A^k could be a tuple in itself (or is it called a latent vector)
                    - Ex. overall RMS first value

- Other values: 0-500hz, 500-2000hz, etc. GETS BA AWESOME LOW LEVEL MID LEVEL HIGH LEVEL BASS MID TREBLE GANON
- f^k <- f0
- Extra parameter: velocity? (is that relevant)

(3) An FM oscillator bank $S_p(.)$ renders a window of N audio samples from output levels $\hat{ol}_k$, fundamental frequency f_{0k}. We configure the bank with the oscillator routing and frequency ratios of the patch p used to train g_ϕ, although this can be changed during inference.

$$s_{Nk}, \ldots s_{N(k+1)} = S_p(\hat{ol}_k, f_{0k}) \quad (4)$$

-
- Where DDSP synthesizer can be used?

- Harmonic synthesizer ← implemented by DDSP
  - Idk

We use Pytorch as a training framework. The process takes about four hours per model using a single NVIDIA GeForce RTX 2080 Ti GPU.

Dataset Preparation

The dataset to be used will be the NSynth dataset. More details.

The following audio features will be used:

1. F0 using CREPE
2. Loudness using STFT and A-weighting

F0 and loudness allows for providing the instantaneous fundamental frequency and intensity of a note sequence at a constant frame rate.

Comparative Analysis

The researchers will examine the existing DDSP-based methods focusing on one-shot neural audio synthesis, such as:

1. Differentiable Wavetable Synthesis by Shan et al. (2021)
2. Neural Instrument Cloning From Very Few Samples by Jonason and Sturm (2022)

DDSP-based methods using external components like synthesizers and more on learning parameters were not chosen.

Do we use pre-trained models or retrain them ourselves using the code?

Hybrid Autoencoder-GAN Architecture

Incorporating a GAN into the DDSP autoencoder architecture and creating a hybrid approach would allow the researchers to produce outputs with finer details and textures that could possibly result in a more "realistic" and characterized waveform generation.

non-causal dilated convolution layers

- nondilated input convolution layer
- Dilation factor
- Kernel
- Layer normalization

a differentiable filtered noise synthesizer by simply applying the LTV-FIR filter from above to a stream of uniform noise Yl = HlNl where Nl is the IDFT of uniform noise in domain [-1, 1].

Metrics

Mas maayos na methodology

1. Dataset ← **NSynth** → **Training Tuples**
    a. F0 ← Fundamental frequency using CREPE
    b. A-weighted Log Amplitude Loudness
    c. Getting “pseudo-envelope” through Hilbert transform (di ko pa sure to)
    d. Z-encoder: using CQT ← constant q transform
2. Data Processing
    a. GAN to refine pitch / F0 for micro-pitch variations???????? (ewan din kasi ginagamit sa vocoder)
    b. Essentially same as DDSP
    c. So many different DDSP variations...
        i. Waveshaping (prediction)
        ii. Subtractive synthesizers...
        iii. FM tone transfer
        iv. Wavetables
3. Metrics
    a. Subjective:
        i. MOS (Mean Opinion Score) ← Rating 1-5
        ii. ABX Testing
    b. Quantitative:
        i. Frechet Audio Distance
        ii. Number of Statistically-Different Bins (NDB)

5.1 Quantitative Results

As a quantitative measure of the quality of sound match, we use log-spectral distortion (*LSD*) and the multi-scale spectral loss (*Multi*). We also calculate the L1 parameter loss (*Param*) for in-domain sounds. The results are shown