GANSYNTH: 
ADVERSARIAL NEURAL AUDIO SYNTHESIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Efficient audio synthesis is an inherently difficult machine learning task, as human perception is sensitive to both global structure and fine-scale waveform coherence. Autoregressive models, such as WaveNet, model local structure at the expense of global latent structure and slow iterative sampling, while Generative Adversarial Networks (GANs), have global latent conditioning and efficient parallel sampling, but struggle to generate locally-coherent audio waveforms. Herein, we demonstrate that GANs can in fact generate high-fidelity and locally-coherent audio by modelling log magnitudes and instantaneous frequencies with sufficient frequency resolution in the spectral domain. Through extensive empirical investigations on the NSynth dataset, we demonstrate that GANs are able to outperform strong WaveNet baselines on automated and human evaluation metrics, and efficiently generate audio $\sim$54,000 times faster than their autoregressive counterparts.

1 INTRODUCTION

Neural audio synthesis, training generative models to efficiently produce audio with both high-fidelity and global structure, is a challenging open problem as it requires modeling temporal scales over at least five orders of magnitude ($\sim$0.1ms to $\sim$100s). Large advances in the state-of-the art have been pioneered almost exclusively by autoregressive models, such as WaveNet, which solve the scale problem by focusing on the finest scale possible (a single audio sample) and rely upon external conditioning signals for global structure (van den Oord et al., 2016). This comes at the cost of slow sampling speed, since they rely on inefficient iterative ancestral sampling to generate waveforms one sample at a time. Due to their high quality, a lot of research has gone into speeding up generation, but the methods introduce significant overhead such as training a secondary student network or writing highly customized low-level kernels (van den Oord et al., 2017; Paine et al., 2016). Furthermore, since these large models operate at a fine timescale, their autoencoder variants are restricted to only modeling local latent structure due to memory constraints (Engel et al., 2017).

On the other end of the spectrum, Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) have seen great recent success at generating high resolution images (Radford et al., 2016; Arjovsky et al., 2017; Gulrajani et al., 2017; Berthelot et al., 2017; Kodali et al., 2017; Karras et al., 2018; Miyato et al., 2018). Typical GANs achieve both efficient parallel sampling and global latent control by conditioning a stack of transposed convolutions on a latent vector. The potential for audio GANs extends further, as adversarial costs have unlocked intriguing domain transformations for images that could possibly have analogues in audio (Isola et al., 2017; Zhu et al., 2017; Wolf et al., 2017; Jin et al., 2017). However, attempts to adapt image GAN architectures to generate waveforms in a straightforward manner fail to reach the same level of perceptual fidelity as their image counterparts (Donahue et al., 2018).

Unlike images, however, most audio waveforms, such as speech and music, are highly periodic. Convolutional filters trained for different tasks on this data commonly learn to form logarithmically-scaled frequency selective filter banks spanning the range of human hearing (Dieleman & Schrauwen, 2014; Zhu et al., 2016). Human perception is also highly sensitive to discontinuities and irregularities in periodic waveforms, so maintaining the regularity of periodic signals over short to intermediate timescales (1ms - 100ms) is crucial. Figure 1 shows that the stride of the frames does not exactly equal a waveform’s periodicity the alignment (Phase) of the two processes over time. This condition is assured as at any time there are typically many different frequencies in
Figure 1: Frame-based estimation of audio waveforms. Much of sound is made up of locally-coherent waves with a given frequency, pictured as the red-yellow sinusoid with black dots at the start of each cycle. The output layer, whether it be a transposed convolution or STFT, has a given size and stride, here depicted as equal with boundaries at the dotted lines. The alignment between the two (Phase, indicated by the solid black line and yellow boxes), precesses in time since the periodicity of the audio and the output stride are not exactly the same. Transposed convolutional filters thus have the difficult task of covering both all the necessary frequencies and all possible phase alignments to preserve phase coherence. For an STFT, we can unroll the phase over the $2\pi$ boundary (orange boxes) and take its derivative to get the instantaneous radial frequency (red boxes), which expresses the constant relationship between audio frequency and frame frequency. The structure for spectra is apparent below where we show each representation for an example trumpet from the NSynth dataset.

![Waveform Diagram](waveform_diagram.png)

In this paper, we investigate the interplay of architecture and representation in synthesizing coherent audio with GANs. Our key findings include:

- Generating log-magnitude spectrograms and phases directly with GANs can produce much more coherent waveforms than directly generating waveforms with strided convolutions.
- Estimating IF spectra leads more coherent audio still than estimating phase.
Frequency resolution matters. Harmonic frequencies are multiples of the fundamental, so low pitches have very tightly spaced harmonics, which can cause blurring and overlap. Both increasing the STFT frame size and switching to mel frequency scale improve performance by creating more separation between the lower harmonic frequencies that are especially important to perception.

On the NSynth dataset, GANs can outperform a strong WaveNet baseline in automatic and human evaluations, and generate samples ∼54,000 times faster.

Global conditioning on latent and pitch vectors allow GANs to generate perceptually smooth interpolation in timbre, and consistent timbral identity across pitch.

2 Experimental Details

2.1 Dataset

We focus our study on the NSynth dataset, which contains 300,000 musical notes from 1,000 different instruments aligned and recorded in isolation. Designed to be a “CelebA of audio”, NSynth is a difficult dataset composed of highly diverse timbres and pitches, but it is also highly structured with labels for pitch, velocity, instrument, and acoustic qualities. Each sample is four seconds long, and sampled at 16kHz, giving 64,000 dimensions. As we wanted to included human evaluations on audio quality, we restricted ourselves to training on the subset of acoustic instruments and fundamental pitches ranging from MIDI 24-84 (∼32-1000Hz), as those timbres are most likely to sound natural to an average listener. This left us with 70,379 samples from instruments that are mostly strings, brass, woodwinds, and mallets. Created a new test/train 80/20 split from shuffled data, as the original split was divided along instrument type, which isn’t desirable for this task.

2.2 Architecture and Representations

Given the recent successes of the architecture presented by Karras et al. (2018) for image modeling, we directly adapt an open source implementation in TensorFlow to produce audio spectra instead of images. While we search over a variety of hyperparameter configurations and learning rates, the fundamental architecture remained unchanged so we direct readers to the original paper for an in-depth analysis, and the appendix for more details.

Briefly, the model samples a random vector $z$ from a spherical Gaussian, and runs it through a stack of transposed convolutions to upsample and generate output data $x = G(z)$, which is fed into a discriminator network of downsampling convolutions (whose architecture mirrors the generator’s) to estimate a divergence measure between the real and generated distributions. As in Karras et al. (2018), we use a gradient penalty to promote Lipschitz continuity, and pixel normalization at each layer. We also try training both progressive and non-progressive variants, and see comparable quality in both. While it is not essential for success, we do see with slightly better convergence time and sample diversity for progressive training, so for the remainder of the paper, all models are compared with progressive training.

The largest difference to the original implementation is that we add pitch conditioning a la AC-GAN (Odena et al. [2017]), where we append a one-hot pitch conditioning label to the latent vector before generation and an auxiliary classification loss in the discriminator that tries to predict the pitch label.

For spectral representations, we compute STFT magnitudes and phase angles using TensorFlow’s built-in implementation. We use an STFT with 256 stride and 1024 frame size, resulting in 75% frame overlap and 513 frequency bins. We trim the Nyquist frequency and pad in time to get an “image” of size (256, 512, 2). We take the log of the magnitude to better constrain the range and then scale the magnitudes to be between -1 and 1 to match the tanh output nonlinearity of the generator network. The phase angle is also scaled to between -1 and 1 and we refer to these variants as “phase” models. We optionally unroll the phase angle and take the finite difference as in Figure 1; we call the resulting models “instantaneous frequency” ("IF") models. We also find performance is sensitive to having sufficient frequency resolution at the lower frequency range. Maintaining 75%
overlap we are able to double the STFT frame size and stride, resulting in spectral images with size \((128, 1024, 2)\), which we refer to as “\(+H\)”, variants. Lastly, to provide even more separation of lower frequencies we transform both the log magnitudes and instantaneous frequencies to a mel frequency scale without dimensionless compression \((1024\) bins), which we refer to as “\(IF-Mel\)” variants. To convert back to linear STFTs we just use the inverse linear transformation, which, perhaps surprisingly does not harm audio quality significantly.

It is important for us to compare against strong baselines, so we adapt WaveGAN \cite{Donahue:2018}, the current state of the art in waveform generation with GANs, to accept pitch conditioning and retrain it on our subset of the NSynth dataset. We also independently train our own waveform generating GANs off the progressive codebase and our best models achieve similar performance to WaveGAN without progressive training, so we opt to primarily show numbers from WaveGAN instead.

Beyond GANs, WaveNet \cite{van-den-Oord:2016} is currently the state of the art in generative modeling of audio. Prior work on the NSynth dataset used an WaveNet autoencoder to interpolate between sounds \cite{Engel:2017}, but is not a generative model as it requires conditioning on the original audio. Thus, we create strong WaveNet baselines by adapting the architecture to accept the same one-hot pitch conditioning signal as the GANs. We train variants using both a categorical 8-bit \(\mu\) law and 16-bit mixture of logistics for the output distributions, but find that the 8-bit model is more stable and outperforms the 16-bit model.

### 3 Metrics

Evaluation generative models is itself a difficult problem: because our goals (perceptually-realistic audio generation) are hard to formalize, the most common evaluation metrics tend to be heuristic and have “blind spots.” \cite{Theis:2016} To mitigate this, we evaluate all of our models against a diverse set of metrics, each of which captures a distinct aspect of model performance. Our evaluation metrics are as follows:

- **Human Evaluation** We use human evaluators as our gold standard of audio quality because it is notoriously hard to measure in an automated manner. In the end, we are interested in training network to synthesize coherent waveforms, specifically because human perception is extremely sensitive to phase irregularities and these irregularities are very disruptive to a listener. We used Amazon Mechanical Turk to perform a comparison test on samples from all models presented in \cite{Karras:2018} (this includes the hold-out dataset). The participants were presented with two 4s samples corresponding to the same pitch. On a Likert scale, the participants answer the question of “which sample has better audio quality / has less audio distortions?”. For the study, we collected 3600 ratings, with each model being involved in 800 comparisons.

- **Number of Statistically-Different Bins (NDB)** We adopt the metric proposed by \cite{Richardson:2018} to measure the diversity of generated samples: the training examples are clustered into \(k = 50\) Voronoi cells by \(k\)-means in log-spectrogram space, the generated examples are also mapped into the same space and are assigned to the nearest cell. NDB is reported as the number of cells where the number of training examples is statistically significantly different from the number of generated examples by a two-sample Binomial test.

- **Inception Score (IS)** \cite{Salimans:2016} propose a metric for evaluating GANs which has become a de-facto standard in GAN literature \cite{Gulrajani:2017,Miyato:2018,Karras:2018}. Generated samples are run through a pretrained Inception classifier and the Inception Score is defined as the mean KL divergence between the image-conditional output class probabilities and the marginal distribution of the same. IS penalizes models whose samples aren’t each easily classified into a single class, as well as models whose samples collectively belong to only a few of the possible classes. Though we still call our metric “IS” for consistency, we replace the Inception features with features from a pitch classifier trained on spectrograms of our acoustic NSynth dataset, since Inception features are unlikely to be semantically meaningful for our task.
• **Pitch Accuracy (PA) and Pitch Entropy (PE)**  Because the Inception Score can conflate models which don’t produce very distinct pitches and models which produce only a few pitches, we also separately measure the accuracy of the same pretrained pitch classifier on generated samples (PA) and the entropy of its output distribution (PE).

• **Fréchet Inception Distance (FID)**  [Heusel et al. 2017] propose a metric for evaluating GANs based on the 2-Wasserstein (or Fréchet) distance between multivariate Gaussians fit to features extracted from a pretrained Inception classifier and show that this metric correlates with perceptual quality and diversity on synthetic distributions. As with Inception Score, we use pitch-classifier features instead of Inception features.

## 4 Results

Table 1: Metrics for different models. “+ H” stands for higher frequency resolution, and “Real Data” is drawn from the test set.

<table>
<thead>
<tr>
<th>Samples</th>
<th>Human Eval</th>
<th>NDB</th>
<th>FID</th>
<th>IS</th>
<th>PA</th>
<th>PE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(wins)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real Data</td>
<td>549</td>
<td>2.2</td>
<td>13</td>
<td>47.1</td>
<td>98.2</td>
<td>0.22</td>
</tr>
<tr>
<td>IF-Mel + H</td>
<td>485</td>
<td><strong>29.3</strong></td>
<td>167</td>
<td>38.1</td>
<td>97.9</td>
<td>0.40</td>
</tr>
<tr>
<td>IF + H</td>
<td>308</td>
<td>36.0</td>
<td><strong>104</strong></td>
<td><strong>41.6</strong></td>
<td><strong>98.3</strong></td>
<td><strong>0.32</strong></td>
</tr>
<tr>
<td>Phase + H</td>
<td>225</td>
<td>37.6</td>
<td>592</td>
<td>36.2</td>
<td>97.6</td>
<td>0.44</td>
</tr>
<tr>
<td>IF-Mel</td>
<td>479</td>
<td>37.0</td>
<td>600</td>
<td>29.6</td>
<td>94.1</td>
<td>0.63</td>
</tr>
<tr>
<td>IF</td>
<td>283</td>
<td>37.0</td>
<td>708</td>
<td>36.3</td>
<td>96.8</td>
<td>0.44</td>
</tr>
<tr>
<td>Phase</td>
<td>203</td>
<td>41.4</td>
<td>687</td>
<td>24.4</td>
<td>94.4</td>
<td>0.77</td>
</tr>
<tr>
<td>WaveNet</td>
<td>359</td>
<td>45.9</td>
<td>320</td>
<td>29.1</td>
<td>92.7</td>
<td>0.70</td>
</tr>
<tr>
<td>WaveGAN</td>
<td>216</td>
<td>43.0</td>
<td>461</td>
<td>13.7</td>
<td>82.7</td>
<td>1.40</td>
</tr>
</tbody>
</table>

Table 1 presents a summary of our results on all model and representation variants. Our most discerning measure of audio quality, human evaluation, shows a very clear trend, with decreasing quality as output representation moves from IF-Mel, IF, Phase, to Waveform. The highest quality model, IF-Mel, was judged comparably but slightly inferior to real data. The WaveNet baseline produces very high-fidelity sounds, but occasionally break down into feedback and self oscillation, resulting in a score that is comparable to the IF GANs.

While there is no a priori reason that sample diversity should correlate with audio quality, we indeed find that NDB follows the same trend as the human evaluation. Additionally, high frequency resolution improves the NDB score across models types. The WaveNet baseline receives the worst NDB score. Even though the generative model assigns high likelihood to all the training data, the autoregressive sampling itself has a tendency gravitate to the same type of oscillation for each given pitch conditioning, leading to an extreme lack of diversity. Histograms of the sample distributions showing peaky distributions for the different models can be found in the appendix.

FID provides a similar story to the first two metrics with significantly lower scores for for IF models with high frequency resolution. Comparatively, Mel scaling has much less of an effect on the FID then it does in the listener study. Phase models have high FID, even at high frequency resolution, reflecting their poor sample quality.

Many of the models do quite well on the classifier metrics of IS, Pitch Accuracy, and Pitch Entropy, because they have explicit conditioning telling them what pitches to generate. All of the high-resolution models actually generate samples classified with very similar accuracy to the real data. As this accuracy and entropy can be a strong function of the distribution of generated samples, which most certainly does not match the training distribution due to mode collapse and other issues, there is very little discriminative information to gain about sample quality from differences among such high scores. The metrics do provide a rough measure of which models are less reliably generating classifiable pitches, which seems to be the low frequency models to some extent and the baselines.
Figure 2: Phase coherence. Samples are selected to be roughly similar between the models for illustrative purposes. The top row shows the waveform modulo the fundamental periodicity of the note (MIDI C60), for 1028 samples taken in the middle of the note. Notice that the real data completely overlaps itself as the waveform is extremely periodic. The WaveGAN and PhaseGAN, however, have many phase irregularities, creating a blurry web of lines. The IFGAN is much more coherent, having only small variations from cycle-to-cycle. In the Rainbowgrams below, the real data and IF models have coherent waveforms that result in strong consistent colors for each harmonic, while the PhaseGAN has many speckles due to phase discontinuities, and the WaveGAN model is very irregular.

5 Qualitative Analysis

While we do our best to visualize qualitative audio concepts, we highly recommend the reader to listen to the accompanying audio samples provided at [https://storage.googleapis.com/gansynth/samples.zip](https://storage.googleapis.com/gansynth/samples.zip).

5.1 Phase Coherence

Figure 2 visualizes the phase coherence of samples from different GAN variants. It is clear from the waveforms at the top, which are wrapped at the fundamental frequency, that the real data and IF models produce waveforms that are very consistent from cycle-to-cycle. The PhaseGAN has some phase discontinuities, while the WaveGAN is very irregular. Below we use Rainbowgrams (Engel et al., 2017) to depict the log magnitude of the frequencies as brightness and the IF as the color on a rainbow color map. This visualization helps to see clear phase coherence of the harmonics in the real data and IFGAN by the strong consistent colors. In contrast, the PhaseGAN discontinuities appear as speckled noise, and the WaveGAN appears largely incoherent.

5.2 Interpolation

As discussed in the introduction, GANs also allow conditioning on the same latent vector the entire sequence, as opposed to only short subsequences for memory intensive autoregressive models like WaveNet. WaveNet autoencoders, such as ones in (Engel et al., 2017), learn local latent codes that control generation on the scale of milliseconds but have limited scope, and have a structure of their own that must be modelled and does not fit a compact prior. In Figure 3, we take a pretrained
Figure 3: Global interpolation. Samples available for download. Interpolating between waveforms perceptually results in crossfading the volumes of two distinct sounds (rainbowgrams at top). The WaveNet autoencoder (middle) only has local conditioning distributed in time, and no compact prior over those time series, so linear interpolation ventures off the true prior / data manifold, and produces in-between sounds that are less realistic samples and feature the default failure mode of autoregressive wavenets (feedback harmonics). Meanwhile, the IF-Mel GAN (bottom) has global conditioning so interpolating in perceptual attributes while staying along the prior at all intermediate samples, so they are high-fidelity audio samples like the endpoints.

WaveNet autoencoder\[^2\] and compare interpolating between samples in the raw waveform (top), the distributed latent code of a WaveNet autoencoder, and the global code of an IF-Mel GAN.

Interpolating the waveform is perceptually equivalent to mixing between the amplitudes of two distinct sounds. WaveNet improves upon this for the two notes by mixing in the space of timbre, but the linear interpolation does not correspond to the complex prior on latents, and the intermediate sounds have a tendency to fall apart, oscillate and whistle, which are the natural failure modes for a WaveNet model. However, the GAN model has a spherical gaussian prior for the entire sound, so spherical interpolation stays well aligned with the prior so the perceptual change is smooth and all intermediate points are valid instrument sounds in their own right.

5.3 Consistent Timbre Across Pitch

While timbre slightly varies for a natural instrument across register, on the whole it remains consistent, giving the instrument it’s unique character. In the downloadable audio samples\[^5\] we fix the latent conditioning variable and generate samples by varying the pitch conditioning over five octaves. It’s clear that the timbral identity of the GAN remains largely intact, creating a unique instrument identity for the given point in latent space.

\[^2\]https://github.com/tensorflow/magenta/tree/master/magenta/models/nsynth
\[^5\]https://storage.googleapis.com/gansynth/samples.zip
6 Fast Generation

One of the advantages of GANs with upsampling convolutions over autoregressive models is that the both the training and generation can be processed in parallel for the entire audio sample. This is quite amenable to modern GPU hardware which is often I/O bound with iterative autoregressive algorithms. This can be seen when we synthesize a single four second audio sample on a TitanX GPU and the latency to completion drops from 1077.53 seconds for the WaveNet baseline to 20 milliseconds for the IF-Mel GAN making it around 53,880 times faster. Previous applications of WaveNet autoencoders trained on the NSynth dataset for music performance relied on prerendering all possible sounds for playback due to the long synthesis latency\[^6\]. This work opens up the intriguing possibility for realtime neural network audio synthesis on device, allowing users to explore a much broader palette of expressive sounds.

7 Related Work

Much work on deep generative models of audio tends to focus on speech synthesis (Sotelo et al., 2017; Wang et al., 2017). In comparison, audio generation for music is relatively under-explored. van den Oord et al. (2016) and Mehri et al. (2017) propose autoregressive models and demonstrate their ability to synthesize musical instrument sounds, but these suffer from the aforementioned slow generation. Donahue et al. (2018) first applied GANs to audio generation with coherent results, but fell short of the audio fidelity of autoregressive likelihood models.

Our work also builds on multiple recent advances in GAN literature. Gulrajani et al. (2017) propose a modification to the loss function of GANs and demonstrate improved training stability and architectural robustness. Karras et al. (2018) further introduce progressive training, in which successive layers of the generator and discriminator are learned in a curriculum, leading to improved generation quality given a limited training time. They also propose a number of architectural tricks to further improve quality, which we employ in our best models.

The NSynth dataset was first introduced as a “CelebA of audio” (Liu et al., 2015; Engel et al., 2017). The original paper focused on WaveNet autoencoders and spectral autoencoders, but did not include any truly generative models because it was not possible to sample from the prior. Follow up work used the dataset and architecture to show fundamental pitch preserving timbre transformation, but suffers the same limitations as the original model (Mor et al., 2018).

8 Conclusion

We have demonstrated high-quality audio generation with GANs on the NSynth dataset, exceeding the fidelity of a strong WaveNet baseline while generating samples tens of thousands of times faster. While this is a major advance for audio generation with GANs, this study focused on a specific controlled dataset, and further work is needed to validate and expand it to a broader class of signals including speech and other types of natural sound. This work also opens up possible avenues for domain transfer and other exciting applications of adversarial losses to audio. Issues of mode collapse and diversity common to GANs exist for audio as well, and we leave it to further work to consider combining adversarial losses with encoders or more straightforward regression losses to better capture the full data distribution.

References


\[^6\]http://g.co/nsynthsuper


A MODEL ABLATION VIA HUMAN STUDY

Figure 4: Number of wins on pair-wise comparison across different output representations and baselines. Higher scores represent better perceptual quality to participants. The ranking observed here correlates well with the evaluation on quantitative metrics as in Table 1.
B Measuring Diversity Across Generated Samples

Figure 5: NDB bin proportions for different models (evaluated with samples of pitch 60).