Adversarial Generation of Time-Frequency Features
with application in audio synthesis

Andrés Marafioti [1] **Nicki Holighaus** [1] **Nathanaël Perraudin** [2] **Piotr Majdak** [1]

Abstract

Time-frequency (TF) representations provide powerful and intuitive features for the analysis of time series such as audio. But still, generative modeling of audio in the TF domain is a subtle matter. Consequently, neural audio synthesis widely relies on directly modeling the waveform and previous attempts at unconditionally synthesizing audio from neurally generated invertible TF features still struggle to produce audio at satisfying quality. In this article, focusing on the short-time Fourier transform, we discuss the challenges that arise in audio synthesis based on generated invertible TF features and how to overcome them. We demonstrate the potential of deliberate generative TF modeling by training a generative adversarial network (GAN) on short-time Fourier features. We show that by applying our guidelines, our TF-based network was able to outperform a state-of-the-art GAN generating waveforms directly, despite the similar architecture in the two networks.

1. Introduction

Despite the recent advance in machine learning and generative modeling, synthesis of natural sounds by neural networks remains a challenge. Recent solutions rely on, among others, classic recurrent neural networks (e.g., SampleRNN, Mehri et al., 2017), dilated convolutions (e.g., WaveNet, Van Den Oord et al., 2016), and generative adversarial networks (e.g., WaveGAN, Donahue et al., 2019). Especially, the latter offers a promising approach in terms of flexibility and quality. Generative adversarial networks (GANs, Goodfellow et al., 2014) rely on two competing neural networks trained simultaneously in a two-player min-max game: The generator produces new data from samples of a random variable; The discriminator attempts to distinguish between these generated and real data. During the training, the generator's objective is to fool the discriminator, while the discriminator attempts to learn to better classify real and generated (fake) data. Since their introduction, GANs have been improved in various ways (e.g., Arjovsky et al., 2017; Gulrajani et al., 2017). For images, GANs have been used to great success (Karras et al., 2018; Brock et al., 2019). For audio, GANs enable the generation of a signal at once even for durations in the range of seconds (Donahue et al., 2019).

The neural generation of realistic audio remains a challenge, because of its complex structure, with dependencies on various temporal scales. In order to address this issue, a network generating audio is often complemented with another neural network or prior information. For example, the former may require a system of two parallel neural networks (Van Den Oord et al., 2018), leading overall to more complex systems, while the latter can take the form of a separate conditioning of networks (Shen et al., 2018; Sotelo et al., 2017; Engel et al., 2017). It is usually beneficial to train neural networks on a high-level representation of sound, instead on the time-domain samples. For example, Tacotron 2 (Shen et al., 2018) relies on non-invertible mel-frequency spectrograms. Generation of a time-domain signal from the mel coefficients is then achieved by training a conditioned WaveNet to act as a vocoder.

Time-frequency (TF) domain representations of sound are successfully used in many applications and rely on well-understood theoretical foundations. They have been widely applied to neural networks, e.g., for solving discriminative tasks, in which they outperform networks directly trained on the waveform (Dieleman & Schrauwen, 2014; Pons et al., 2017). Further, TF representations are used to parameterize neural synthesizers, e.g., Tacotron 2 mentioned above or Timbretron (Huang et al., 2019), which modifies timbre by remapping constant-Q TF coefficients of sound, conditioning a WaveNet synthesizer. Despite the success of TF representations for sound *analysis*, why, one could ask, has neural *sound generation* via invertible TF representations only seen limited success?

---

[1]Acoustics Research Institute, Austrian Academy of Sciences, Wohllebengasse 12–14, 1040 Vienna, Austria. [2]Swiss Data Science Center, ETH Zürich, Universitätstrasse 25, 8006 Zürich. Correspondence to: Andrés Marafioti <amarafioti@kfs.oeaw.ac.at>.

Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

In fact, there *are* neural networks generating invertible TF representations for sound synthesis. They were designed to perform a specific task such as source separation (Fan et al., 2018; Muth et al., 2018), speech enhancement (Pascual et al., 2017), or audio inpainting (Marafioti et al., 2018) and use a specific and well-chosen setup for TF processing. While the general rules for the parameter choice are not the main focus of those contributions, these rules are highly relevant when it comes to synthesizing sound from a set of TF coefficients generated, e.g., by a neural network.

When both the TF representation and its parameters are appropriately chosen, we generate a highly structured, invertible representation of sound, from which time-domain audio can be obtained using efficient, content-independent reconstruction algorithms. In that case, we do not need to train a problem-specific neural synthesizer. Hence, in this article, we discuss important aspects of neural generation of TF representations particularly for sound synthesis. We focus on the short-time Fourier transform (STFT, e.g., Allen, 1977; Wexler & Raz, 1990), the best understood and widely used TF representation in the field of audio processing. First, we revisit some properties of the continuous STFT (Portnoff, 1976; Auger et al., 2012; Gröchenig, 2001) and the progress in phaseless reconstruction of audio signals from STFT coefficients (Průša et al., 2017). Then, we discuss these properties in the context of the discrete STFT in order to compile guidelines for the choice of STFT parameters ensuring the reliability of sound synthesis and to provide tools monitoring the training progress of the generative models. For the latter, we introduce a novel, experimental measure for the consistency of the STFT. Eventually, we demonstrate the applicability of our guidelines by introducing TiFGAN, a network which generates audio using a TF representation. We provide perceptual and numerical evaluations of TiFGAN demonstrating improved audio quality compared to a state-of-the-art GAN for audio synthesis[1]. Our software, complemented by instructive examples, is available at `http://tifgan.github.io`.

2. Properties of the STFT

The rich structure of the STFT is particularly apparent in the continuous setting of square-integrable functions, i.e. functions in $\mathbf{L}^2(\mathbb{R})$. Thus, we first discuss the core issues that arise in the generation of STFTs within that setting, recalling established theory along the way, and then move to discuss these issues in the setting of discrete STFTs.

[1]During the preparation of this manuscript, the work (Engel et al., 2019) became publicly available. In addition to well chosen STFT parameters, usage of the time-direction phase derivative enabled their model, GANSynth, to produce significantly better results than previous methods. The authors kindly provided us with details of their implementation, enabling a preliminary discussion of similarities and differences to our guidelines.

2.1. The continuous STFT

The STFT of the function $f \in \mathbf{L}^2(\mathbb{R})$ with respect to the window $\varphi \in \mathbf{L}^2(\mathbb{R})$ is given by

$$\mathrm{V}_\varphi f(x,\omega) = \int_{\mathbb{R}} f(t)\overline{\varphi(t-x)}\mathrm{e}^{-2\pi i\omega t}\,dt \tag{1}$$

The variable $(x,\omega) \in \mathbb{R}^2$ indicates that $\mathrm{V}_\varphi f(x,\omega)$ describes the time-frequency content of f at time x and frequency ω. The STFT is complex-valued and can be rewritten in terms of two real-valued functions as $\mathrm{V}_\varphi f(x,\omega) = \exp(\mathrm{M}_\varphi(x,\omega) + i\phi_\varphi(x,\omega))$, whenever $\mathrm{V}_\varphi f(x,\omega) \neq 0$. The logarithmic magnitude (*log-magnitude*) M_φ is uniquely defined, but the *phase* ϕ_φ is only defined modulo 2π. Further, while M_φ is a smooth, slowly varying function, ϕ_φ may vary rapidly and is significantly harder to model accurately. Nonetheless, both functions are intimately related. If $\varphi(t) = \varphi_\lambda(t) := e^{-\pi t^2/\lambda}$ is a Gaussian window, this relation can be made explicit (Portnoff, 1976; Auger et al., 2012) through the phase-magnitude relations

$$\begin{aligned} \frac{\partial \phi_{\varphi_\lambda}}{\partial x}(x,\omega) &= \lambda^{-1}\frac{\partial \mathrm{M}_{\varphi_\lambda}}{\partial \omega}(x,\omega), \\ \frac{\partial \phi_{\varphi_\lambda}}{\partial \omega}(x,\omega) &= -\lambda\frac{\partial \mathrm{M}_{\varphi_\lambda}}{\partial x}(x,\omega) - 2\pi x, \end{aligned} \tag{2}$$

where $\frac{\partial}{\partial \bullet}$ denotes partial derivatives with respect to $\bullet$. Hence, as long as we avoid zeros of $\mathrm{V}_\varphi f$, the phase ϕ_{φ_λ} can be recovered from $\mathrm{M}_{\varphi_\lambda}$ up to a global constant. Since the STFT is invertible, we can recover f from $\mathrm{M}_{\varphi_\lambda}$ up to a global phase factor as well, such that it is sufficient to model only the magnitude $\mathrm{M}_{\varphi_\lambda}$.

Note that the partial phase derivatives are of interest by themselves. In contrast to the phase itself, they provide an intuitive interpretation as local instantaneous frequency and time and are useful in various applications (Dolson, 1986; Auger & Flandrin, 1995). Further, as suggested by (2), the phase derivatives might be a more promising modeling target than the phase itself, at least after unwrapping and demodulation[2] as detailed in (Arfib et al., 2011).

Note that not every function $F \in \mathbf{L}^2(\mathbb{R}^2)$ is the STFT of a time-domain signal because the STFT operator V_φ maps $\mathbf{L}^2(\mathbb{R})$ to a strict subspace of $\mathbf{L}^2(\mathbb{R}^2)$. Formally, assuming that the window φ has unit norm, the inverse STFT is given by the adjoint operator V_φ^* of V_φ and we have $\mathrm{V}_\varphi^*(\mathrm{V}_\varphi f) = f$ for all f. Now, if $F \in \mathbf{L}^2(\mathbb{R}^2)$ is not in the range of V_φ, then $f = \mathrm{V}_\varphi^* F$ is a valid time-domain signal, but $F \neq \mathrm{V}_\varphi f$, i.e., F is an *inconsistent* representation of f, and the TF structure of F will be distorted in $\mathrm{V}_\varphi f$.

In the presence of phase information, consistency of F can be evaluated simply by computing the norm difference $\|F - \mathrm{V}_\varphi(\mathrm{V}_\varphi^* F)\|$ which can also serve as part of a training objective. If only magnitudes $\widetilde{\mathrm{M}}$ are available, we can

[2]Formally, demodulation is simply adding $2\pi x$ to $\frac{\partial \phi_{\varphi_\lambda}}{\partial \omega}(x,\omega)$.

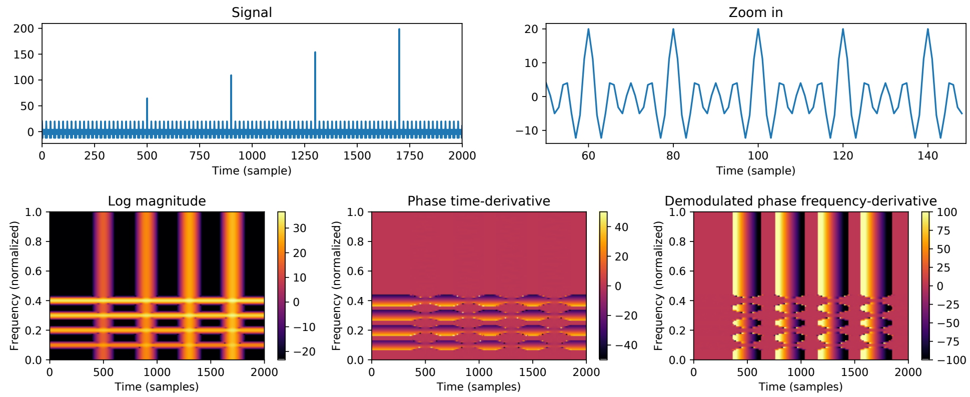

Figure 1. Signal representations. Top row: waveform of a test signal (pure tone and pulses). Bottom row: STFT features: log magnitudes (left), time-direction phase derivatives (center) and frequency-direction phase derivatives (right). For small log magnitude, phase derivatives were set to zero. Frequency-direction derivative was computed after demodulation.

theoretically exploit the phase-magnitude relations (2), reconstruct the phase, and then evaluate consistency. Unless otherwise specified, coefficients that are not necessarily consistent are indicated by the symbol $\sim$, e.g., generated magnitudes $\widetilde{\mathrm{M}}$. In practice, phase recovery from the magnitude $\widetilde{\mathrm{M}}$ introduces errors of its own and the combined process may become too expensive to be attractive as a training objective. Thus, it might be preferable to evaluate consistency of the generated magnitude directly, which, for Gaussian windows, can be derived from (2)

$$\left(\lambda \frac{\partial^2}{\partial x^2} + \lambda^{-1} \frac{\partial^2}{\partial \omega^2}\right) \mathrm{M}_{\varphi_\lambda}(x,\omega) = -2\pi. \tag{3}$$

Note that, although (Portnoff, 1976) already observed that $\widetilde{\mathrm{M}}$ is an STFT magnitude if and only if (3) holds (and $e^{\widetilde{\mathrm{M}}}$ is square-integrable), our contribution is, to our knowledge, the first to exploit this relation to evaluate consistency. Furthermore, the phase-magnitude relations (2) and the consistency equivalence (3) can be traced back to the relation of Gaussian STFTs to a certain space of analytic functions (Bargmann, 1961; Conway, 1973).

In the context of neural networks, the ultimate goal of the generation process is to obtain a time-domain signal, but we can only generate a finite number of STFT coefficients. Therefore, it is essential that inversion from the generated values is possible and synthesis of the time-domain signal is robust to distortions. In mathematical terms, this requires a window function φ and time and frequency steps $a, b \in \mathbb{R}^+$ specifying a snug STFT (or Gabor) *frame* (Christensen, 2016). While a comprehensive discussion of STFT frames is beyond the scope of this article, it is generally advisable to match a, b to the width of φ and its Fourier transform $\hat{\varphi}$. In the case where both φ and $\hat{\varphi}$ are at least remotely bell-shaped, a straightforward measure of their widths are the standard deviations $\sigma_\varphi = \sigma(\varphi/\|\varphi\|_{\mathbf{L}^1})$ and $\sigma_{\hat{\varphi}} = \sigma(\hat{\varphi}/\|\hat{\varphi}\|_{\mathbf{L}^1})$. Hence, we expect good results if $a/b = \sigma_\varphi/\sigma_{\hat{\varphi}}$. For Gaussian windows φ_λ, we have $\sigma_{\varphi_\lambda}/\sigma_{\widehat{\varphi_\lambda}} = \lambda$, such that λ is often referred to as *time-frequency ratio*. For such φ_λ, the choice $a/b = \lambda$ is conjectured to be optimal[3] in general (Strohmer & Beaver, 2003), and proven to be so for $(ab)^{-1} \in \mathbb{N}$ (Faulhuber & Steinerberger, 2017). Furthermore, the relations (2) and (3) only hold exactly for the undecimated STFT and must be approximated. For this approximation to be reliable, ab must be small enough. The theory suggests that $ab \leq 1/4$ is generally required for reliable reconstruction of signals from the magnitude alone (Balan et al., 2006). For larger ab, the values of the STFT become increasingly independent and little exploitable (or learnable) structure remains.

These considerations provide useful guidelines for the choice of STFT parameters. In the following, we translate them into a discrete implementation.

2.2. The discrete STFT

The STFT of a finite, real signal $s \in \mathbb{R}^L$, with the *analysis window* $g \in \mathbb{R}^L$, time step $a \in \mathbb{N}$ and $M \in \mathbb{N}$ frequency channels is given by

$$\mathrm{S}_g(s)[m,n] = \sum_{l \in \underline{L}} s[l] g[l-na] e^{-2\pi i m l/M}, \tag{4}$$

for $n \in \underline{N}$, $m \in \underline{M}$, where we denote, for any $j \in \mathbb{N}$, $\underline{j} = [0, \ldots, j-1]$ and indices are to be understood modulo L. Similar to the continuous case, we can write $\mathrm{S}_g(s)[m,n] = \exp(\mathrm{M}_g[m,n] + i\phi_g[m,n])$, with log-magnitude M_g and phase ϕ_g. The vectors $\mathrm{S}_g(s)[\cdot, n] \in \mathbb{C}^N$ and $\mathrm{S}_g(s)[m, \cdot] \in \mathbb{C}^M$ are called the n-th (time) segment and m-th (frequency) channel of the STFT, respectively.

Let $b = L/M$. Then, the ratio $M/a = L/(ab)$ is a measure of the transform redundancy and the STFT is overcomplete (or redundant) if $M/a > 1$. If s and g are real-valued, all time segments are conjugate symmetric and it is sufficient to store the first $M_{\mathbb{R}} = \lfloor M/2 \rfloor$ channels only, such that the

[3] In the sense of the frame bound ratio, which is a measure of transform stability (Christensen, 2016).

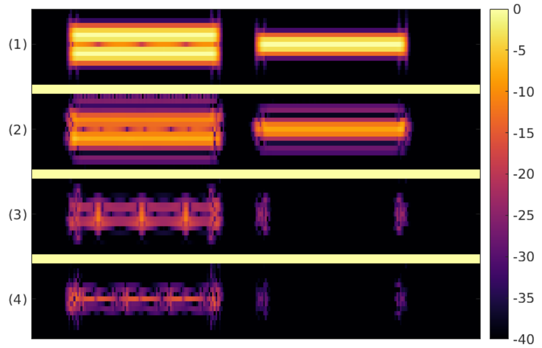

Figure 2. Overview of spectral changes resulting from different phase reconstruction methods. (1) Original log-magnitude, (2-4) log-magnitude differences between original and signals restored with (2) cumulative sum along channels (initialized with zeros), (3) PGHI from phase derivatives (4) PGHI from magnitude only and phase estimated from Eq. (5).

STFT matrix can be reduced to the size $M_{\mathbb{R}} \times N$.

The inverse STFT with respect to the *synthesis window* $\widetilde{g} \in \mathbb{R}^L$ can be written as $\widetilde{s}[l] = \sum_{n\in \underline{N}} \sum_{m\in \underline{M}} \mathrm{S}_g(s)[m,n]\widetilde{g}[l-na]e^{2\pi iml/M}$, $l \in \underline{L}$. We say that $\widetilde{g}$ is a *dual window* for g, if $\widetilde{s} = s$ for all $s \in \mathbb{R}^L$ (Strohmer, 1998; Janssen, 1997; Wexler & Raz, 1990).

Note that the number of channels M can be smaller than the number of nonzero samples L_g of g, as long as a and M respect the widths of g and its Fourier transform $\hat{g}$ as discussed in Sec. 2.1. This yields $aM \approx L\frac{\sigma_g}{\sigma_{\hat{g}}}$ as a general guideline with $\sigma_g = \sigma(g/\|g\|_{\ell^1})$ and $\sigma_{\hat{g}} = \sigma(\hat{g}/\|\hat{g}\|_{\ell^1})$. Furthermore, with the redundancy $M/a \geq 4$, there is sufficient dependency between the values of the STFT, e.g., to facilitate magnitude-only reconstruction. In our experience, this choice represents a lower bound for reliability of discrete approximation of (2).

! Implementations of STFT in SciPy and Tensorflow introduce a phase skew dependent on the (stored) window length L_g (usually $L_g \ll L$) and with severe effects on any phase analysis and processing if not accounted for. This can be addressed with the conversion between (4) and other conventions presented in the supplementary material D and (Arfib et al., 2011; Pruša, 2015).

2.3. Phase recovery and the phase-magnitude relationship

Let $\partial_\bullet$ denote some discrete partial differentiation scheme. Discrete approximation of the phase-magnitude relationship (2) results in

$$\begin{aligned} \partial_n \phi_g[m,n] &\approx \frac{aM}{\lambda}\partial_m \mathbf{M}_g[m,n], \\ \partial_m \phi_g[m,n] &\approx -\frac{\lambda}{aM}\partial_n \mathbf{M}_g[m,n] - 2\pi na/M, \end{aligned} \tag{5}$$

as derived in (Průša et al., 2017). For non-Gaussian windows g, choosing $\lambda = \sigma_g/\sigma_{\hat{g}}$ has shown surprisingly reliable results, but accuracy of (5) depends on the proximity of the window g to a Gaussian nonetheless. Optimal results are obtained for Gaussian windows at redundancy $M/a = L$. While STFTs with $M/a = 4$ perform decently and are considered in our network architecture. In Section 3 we show that a further, moderate increase in redundancy has the potential to further elevate synthesis quality. As an alternative to estimating the phase derivatives from the magnitude, it may be feasible to generate estimates of the phase derivative directly within a generative model.

It may seem straightforward to restore the phase from its time-direction derivative by summation along frequency channels as proposed in (Engel et al., 2019). Even on real, unmodified STFTs, the resulting phase misalignment introduces cancellation between frequency bands resulting in energy loss, see Fig. 2(2) for a simple example. In practice, such cancellations often leads to clearly perceptible changes of timbre[4]. Moreover, in areas of small STFT magnitude, the phase is known to be unreliable (Balazs et al., 2016) and sensitive to distortions (Alaifari & Wellershoff, 2019; Alaifari & Grohs, 2017; Mallat & Waldspurger, 2015), such that it cannot be reliably modelled and synthesis from generated phase derivatives is likely to introduce more distortion. Phase-gradient heap integration (PGHI, Průša et al., 2017) relies on the phase-magnitude relations (5) and bypasses phase instabilities by avoiding integration through areas of small magnitude, leading to significantly better and more robust phase estimates $\widetilde{\phi}$, see Fig. 2(4). PGHI often outperforms more expensive, iterative schemes relying on alternate projection, e.g., Griffin-Lim (Griffin & Lim, 1984; Le Roux et al., 2010; Perraudin et al., 2013), at the phaseless reconstruction (PLR) task. Generally, PLR relies heavily on consistent STFT magnitude for good results. Note that the integration step in PGHI can also be applied if phase derivative estimates from some other source are available, e.g., when training a network to learn time- and frequency-direction phase derivatives. For an example, see Fig. 2(3).

2.4. Consistency of the STFT

The space of valid STFTs with a given window is a lower dimensional subspace of all complex-valued matrices of size $M_{\mathbb{R}} \times N$ and a given, generated matrix $\widetilde{\mathrm{S}}$ may be very far from the STFT of any time-domain signal, even if it *looks* correct. To prevent artifacts, it is important to ensure that $\widetilde{\mathrm{S}}$ is *consistent*. Let $\mathrm{iS}_{\widetilde{g}}$ denote the inverse STFT with the dual window $\widetilde{g}$, see Sec. 2.2. Consistency of $\widetilde{\mathrm{S}}$ can be evaluated by computing the *projection error*

$$\mathrm{e}^{\mathrm{proj}} = \|\widetilde{\mathrm{S}} - \mathrm{S}_g(\mathrm{iS}_{\widetilde{g}}(\widetilde{\mathrm{S}}))\|, \tag{6}$$

[4]See http://tifgan.github.io for examples.

where $\|\cdot\|$ denotes the Euclidean norm. When e^{proj} is large, its effects on the synthesized signal are unpredictable and degraded synthesis quality must be expected. Although e^{proj} is an accurate consistency measure, it can be computationally expensive. Further, its use for evaluating the consistency of magnitude-only data is limited: When preceded by phase recovery, e^{proj} is unable to distinguish the error introduced by the employed PLR method from inconsistency of the provided magnitude data.

As an alternative, we instead propose an experimental measure that evaluates consistency of the log-magnitude directly. The proposed consistency measure exploits the consistency relation (3). An approximation in the spirit of (5) yields

$$\frac{\lambda}{a^2}\partial_n^2 \mathrm{M}_g[m,n] + \frac{M^2}{\lambda}\partial_m^2 \mathrm{M}_g[m,n] \approx -2\pi. \tag{7}$$

In practice, and in particular at moderate redundancy, we found (7) to be prone to approximation errors. Experimentally, however, a measure inspired by the sample Pearson correlation (Lyons, 1991) provided promising results. Let $\widetilde{\mathrm{M}}$ be the generated magnitude, we have

$$\mathrm{DM}_n = |\partial_n^2 \widetilde{\mathrm{M}} + \tfrac{\pi a^2}{\lambda}|, \ \mathrm{DM}_m = |\partial_m^2 \widetilde{\mathrm{M}} + \tfrac{\pi\lambda}{M^2}|, \tag{8}$$

where the terms $\pi a^2/\lambda$ and $\pi\lambda/M^2$ are obtained by distributing the shift 2π in (7) equally to both terms on the left hand side. We define the consistency $\varrho(\widetilde{\mathrm{M}})$ of $\widetilde{\mathrm{M}}$ as

$$\varrho(\widetilde{\mathrm{M}}) := \mathrm{r}(\mathrm{DM}_n, \mathrm{DM}_m), \tag{9}$$

where $\mathrm{r}(X,Y)$ is the sample Pearson correlation coefficient of the paired sets of samples (X,Y). If the equality is satisfied in (7), then $\varrho(\widetilde{\mathrm{M}}) = 1$. Conversely if $\varrho(\widetilde{\mathrm{M}}) \approx 0$, then (7) is surely violated and the representation is inconsistent. The performance of ϱ as consistency measure is discussed in Section 3 below.

3. Performance of the consistency measure

The purpose of the consistency measure ϱ is to determine whether a generated log-magnitude is likely to be close to the log-magnitude STFT of a signal, i.e. it is consistent. As discussed above, consistency is crucial to prevent degraded synthesis quality. Hence, it is important to evaluate the dependence of its properties on changes in the redundancy, the window function and its sensitivity to distortion.

In a first test, we compute the mean and standard deviation of ϱ on a speech and a music dataset, see Section 4 for details, at various redundancies, using Gaussian and Hann windows with time-frequency ratio $\lambda \approx 4$ and STFT parameters satisfying $aM/L = 4$, see Fig. 3. We note that a Gaussian random matrix takes surprisingly high values for ϱ and, thus, ϱ is not reliable below redundancy 4. For

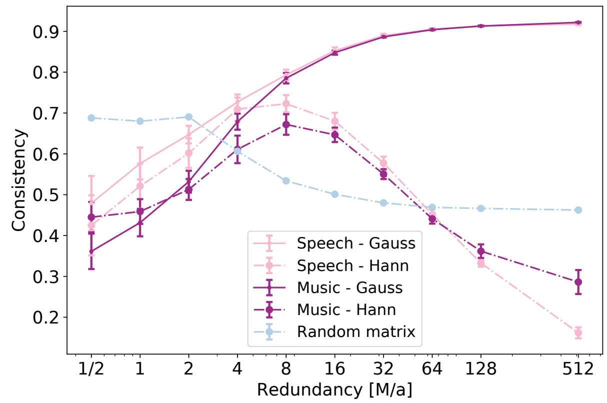

Figure 3. Consistency as function of the redundancy for various time-domain windows. Random matrix from Gaussian distribution.

Gaussian windows, mean consistency increases with redundancy, while the standard deviation decreases, indicating that ϱ becomes increasingly reliable and insensitive to signal content. This analysis suggests that a redundancy of 8 or 16 could lead to notable improvements. At redundancy 4, spectrograms for both types of data score reliably better than the random case, with speech scoring higher than music. The Hann window scores worse than the Gaussian on average in all conditions, with a drop in performance above $M/a = 16$. This indicates that ϱ is only suitable to evaluate consistency of Hann window log-magnitudes for redundancies in the range 6 to 16.

In a second test, we fix a Gaussian STFT with redundancies 4 and 8 and evaluate the behaviour of ϱ under deviations from true STFT magnitudes. To this end, we add various levels of uniform Gaussian noise to the STFT before computing the log–magnitude, see Fig. 4. At redundancy 8 we observe a monotonous decrease of consistency with increasing noise level. In fact, the consistency converges to the level of random noise at high noise levels. Especially for music, ϱ is sensitive to even small levels of noise. At redundancy 4, the changes are not quite as pronounced, but the general trend is similar. While this is not a full analysis of the measure ϱ, it is reasonable to expect that models that match the value of ϱ closely generate approximately consistent log-magnitudes

Furthermore, the results suggest that ϱ has a low standard deviation across data of the same distribution. In the context of GANs, where no obvious convergence criterion applies, ϱ can thus assist the determination of convergence and divergence by tracking

$$\gamma = \left| \mathbb{E}_{\mathbf{M}\sim\mathbb{P}_{\mathbf{M}_{\text{real}}}}\left[\varrho(\mathbf{M})\right] - \mathbb{E}_{\mathbf{M}\sim\mathbb{P}_{\mathbf{M}_{\text{fake}}}}\left[\varrho(\mathbf{M})\right] \right|. \tag{10}$$

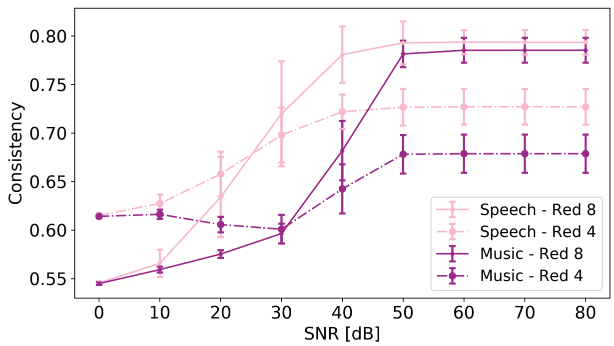

Figure 4. Consistency as a function of SNR obtained by adding complex-valued Gaussian noise to the STFT coefficients.

4. Time-Frequency Generative Adversarial Network (TiFGAN)

To demonstrate the potential of the guidelines and principles for generating short-time Fourier data presented in Section 2, we apply them to TiFGAN, which unconditionally generates audio using a TF representation and improves on the current state-of-the-art for audio synthesis with GANs. For the purpose of this contribution, we restrict to generating 1 second of audio, or more precisely $L = 16384$ samples sampled at 16 kHz. For the short-time Fourier transform, we fix the minimal redundancy that we consider reliable, i.e., $M/a = 4$ and select $a = 128, M = 512$, such that $M_{\mathbb{R}} = 257$, $N = L/a = 128$ and the STFT matrix S is of size $\mathbb{C}^{M_{\mathbb{R}} \times N}$. This implies that the frequency step is $b = L/M = 32$, such that we chose for the analysis window g a (sampled) Gaussian with time-frequency ratio $\lambda = 4 = aM/L$. Since the Nyquist frequency is not expected to hold significant information for the considered signals, we drop it to arrive at a representation size of 256×128, which is well suited to processing using strided convolutions.

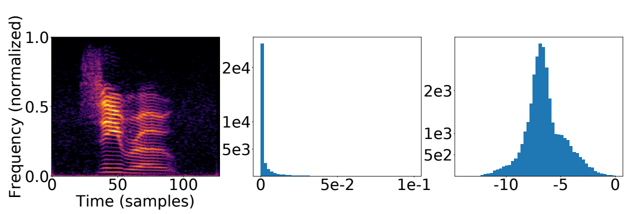

Figure 5. From left to right: log-magnitude spectrogram, distribution of the magnitude, distribution of the log-magnitude.

The log-magnitude distribution is closer to human sound perception and, as show in Fig. 5, it doesn't have the large tail of the magnitude STFT coefficients, therefore we use it for the training data. To do so, we first normalize the STFT magnitude to have maximum value 1, such that the log-magnitude is confined in $(-\infty, 0]$. Then, the dynamic range of the log-magnitude is limited by clipping at $-r$ (in our experiments $r = 10$), before scaling and shifting to the range of the generator output $[-1, 1]$, i.e. dividing

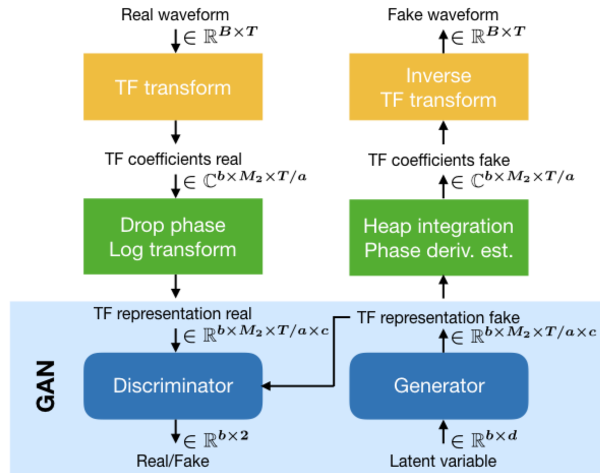

Figure 6. The general architecture with parameters $T = 16384$, $a = 128$, $M_2 = 256$ $c = 1, 3$, $d = 100$. Here $b = 64$ is the batch size. The orange and green steps describe the pre- and post-processing stages.

by $r/2$ before adding constant 1. The network trained to generate log-magnitudes will be referred to as TiFGAN-M. Generation of, and synthesis from, the log-magnitude STFT is the main focus of this contribution. Nonetheless, we also trained a variant architecture TiFGAN-MTF for which we additionally provided the time- and frequency-direction derivatives of the (unwrapped, demodulated) phase[5] (Arfib et al., 2011; Dolson, 1986).

For TiFGAN-M, the phase derivatives are estimated from the generated log-magnitude following (5). For both TiFGAN-M and TiFGAN-MTF, the phase is reconstructed from the phase derivative estimates using phase-gradient heap integration (PGHI, Průša et al., 2017), which requires no iteration, such that reconstruction time is comparable to simply integrating the phase derivatives. For synthesis from the STFT, we use the *canonical dual window* (Strohmer, 1998; Christensen, 2016), precomputed using the Large Time-Frequency Analysis Toolbox (LTFAT, Průša et al., 2014), avalable at `ltfat.github.io`.

GAN architecture: The TiFGAN architecture, depicted in Fig. 6, is an adaptation of DCGAN (Radford et al., 2016) and similarly to WaveGAN and SpecGAN (Donahue et al., 2019), we add one convolutional layer each to generator and discriminator to enable the generation of larger matrices. Moreover, we generate data of size $(256, 128)$, a rectangular array of twice the width and four times the height of DCGANs output, and twice the height of SpecGAN, such that we also adapted the filter shapes to better reflect and cap-

[5]Phase derivatives were obtained using the `gabphasegrad` function in the Large Time-Frequency Analysis Toolbox (LTFAT, Průša et al., 2014).

ture the rectangular shape of the training data[6]. Precisely in comparison to SpecGAN, we use filters of shape $(12, 3)$ instead of the 31% smaller $(5, 5)$. To compensate, we further reduce the number of filter channels of the fully-connected layer and the first convolutional layer of the generator by a factor of 2. Since these two layers comprise the majority of parameters, our architecture only has 10% more parameters than SpecGAN in total. More details on the architecture can be found in Section A of the supplementary material.

Training: During training of TiFGAN, we monitored the relative consistency γ of the generator (10) in addition to the adversarial loss, negative critic and gradient penalty. In the optimization phase, networks that failed to train well, could often be detected to diverge in consistency and discarded after less than $50k$ steps of training (1 day), while promising candidates quickly started to converge towards the consistency of the training data, i.e., $\gamma \to 0$, see Fig. 7. Networks with smaller γ synthesized better audio, but when trained for many steps, they were sometimes less reliable in terms of semantic audio content, e.g., for speech they were more likely to produce gibberish words than with shorter training. Our networks were trained for 200k steps as this seemed to provide reasonably good results in both semantic and audio quality. We optimized the Wasserstein loss (Gulrajani et al., 2017) with the gradient penalty hyperparameter set to 10 using the ADAM optimizer (Kingma & Ba, 2015) with $\alpha = 10^{-4},\ \beta_1 = 0.5,\ \beta_2 = 0.9$ and performed 5 updates of the discriminator for every update of the generator. For the reference condition, we used the pre-trained WaveGAN network provided by (Donahue et al., 2019)[7].

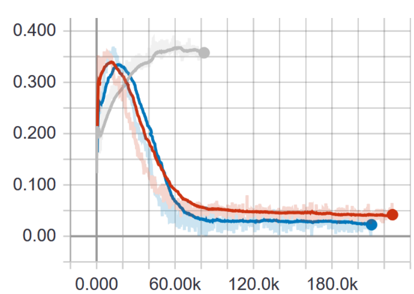

Figure 7. Eq. (10) for three networks. Gray: failed network. Red: TiFGAN-M. Blue: TiFGAN-MTF as in Sec. 4.

Comparison to SpecGAN (Donahue et al., 2019): TiFGAN is purposefully designed to be similar to SpecGAN[8] to emphasize that the main cause for improved results is the handling of time-frequency data according to the guidelines in Section 2.2. SpecGAN relies on an STFT of redundancy $M/a = 2$ with Hann window of length $L_g = 256$, time step $a = 128$ and $M = 256$ channels. According to Section 2.2, this setup is not very well suited to generative modeling. PLR in particular is expected to be unreliable, which is evidenced by the results reported in (Donahue et al., 2019), which employ the classical Griffin-Lim algorithm (Griffin & Lim, 1984) for PLR. The choice of STFT parameters for SpecGAN fixes a target size of shape $(128, 128)$, while for TiFGAN the target size is $(256, 128)$. This required some changes to the network architecture, as presented above. Finally, SpecGAN performs a normalization per frequency channel over the entire dataset, preventing the network to learn the natural relations between channels in the STFT log-magnitude, which are crucial for consistency, as shown in Section 2.2.

4.1. Evaluation

To evaluate the performance of TiFGAN, we trained TiFGAN-M and TiFGAN-MTF using the procedure outlined above on two datasets from (Donahue et al., 2019): (a) Speech, a subset of spoken digits "zero" through "nine" (sc09) from the "Speech Commands Dataset" (Warden, 2018). This dataset is not curated, some samples are noisy or poorly labeled, the considered subset consists of approximately 23,000 samples. (b) Music, a dataset of 25 minutes of piano recordings of Bach compositions, segmented into approximately 19,000 overlapping samples of 1 s duration.

Evaluation metrics: For speech and music, we provide audio examples online[9]. For speech, we performed listening tests and evaluated the inception score (IS) (Salimans et al., 2016) and Fréchet inception distance (FID) (Heusel et al., 2017), using the pre-trained classifier provided with (Donahue et al., 2019). For the real data and both variants of TiFGAN, we moreover computed the consistency ϱ, see Eq. (9), and the relative spectral projection error (RSPE) in dB, after phase reconstruction from the log-magnitude, i.e.,

$$10 \log_{10} \left(\frac{\| |\widetilde{\mathrm{S}}| - |\,\mathrm{S}_g(\mathrm{iS}_{\widetilde{g}}(\widetilde{\mathrm{S}}))| \|}{\|\widetilde{\mathrm{S}}\|} \right), \tag{11}$$

where $|\widetilde{\mathrm{S}}| = |\,\mathrm{S}_g(s)|$ in the case of real data and $|\widetilde{\mathrm{S}}| = \exp(\widetilde{\mathrm{M}})$, with the generated log-magnitude $\widetilde{\mathrm{M}}$, for the generated data. Phase-gradient heap integration was applied to obtain $\widetilde{\mathrm{S}}$ from $|\widetilde{\mathrm{S}}|$ (and generated phase derivatives in the case of TiFGAN-MTF).

Listening tests were performed in a sound booth and sounds were presented via headphones, see supplementary material B. The task involved pairwise comparison of preference between four conditions: real data extracted from the dataset,

[6] When training on piano data, we also observed that, when using square filters, the frequency content of note onsets was unnaturally dispersed over time. This effect was notably reduced after switching to tall filters

[7] https://github.com/chrisdonahue/wavegan

[8] Note that SpecGAN is of equal size as WaveGAN.

[9] http://tifgan.github.io

	vs TiFGAN-M	vs TiFGAN-MTF	vs WaveGAN	Cons	RSPE (dB)	IS	FID
Real	86%	90%	94%	0.70	-22.0*	7.98	0.5
TiFGAN-M	–	67%	75%	0.67	-13.8	5.97	26.7
TiFGAN-MTF	33%	–	55%	0.68	-12.5*	4.48	32.6
WaveGAN	25%	45%	–	–	–	4.64	41.6

Table 1. Results of the evaluation. First three left columns: Preference (in %) of the condition shown in a row over the conditions show in a column, obtained from listening tests. Cons: averaged consistency measure ρ. RSPE: as in Eq. (11). IS: inception score. FID: Fréchet inception distance. *These values were obtained by discarding the phase and reconstructing from the magnitude only. For the listening tests, the signals contained the full representation.

TiFGAN-M generated examples, TiFGAN-MTF generated examples, and WaveGAN generated examples. In each trial, listeners were provided with two sounds from two different conditions. The question to the listener was "which sound do you prefer?". Signals were selected at random from 600 pre-generated examples per condition. Each of the six possible combinations was repeated 80 times in random order, yielding 480 trials per listener. The test lasted approximately 45 minutes including breaks which subjects were allowed to take at any time. Seven subjects were tested and none of them were the authors. A post-screening showed that one subject was not able to distinguish between any of the conditions and thus was removed from the test, yielding in 2880 valid preferences in total from six subjects.

Results: The results are summarized in Table 1. On average, the subjects preferred the real samples over WaveGAN's in 94% of the examples given. For TiFGAN-MTF, the preference decreased to 90% and for TiFGAN-M further to 86%. The large gap between generated and real data can be explained by the experimental setup that enables a very critical evaluation. Nonetheless, it is apparent that TiFGAN-M performed best in the direct comparison to real data by a significant margin. Comparison of the other pairings leads to a similar conclusion: Subjects preferred TiFGAN-MTF over WaveGAN in 55% of the examples given, TiFGAN-M over WaveGAN in 75% and TiFGAN-M over TiFGAN-MTF in 67%. While TiFGAN-M clearly outperformed the other networks, TiFGAN-MTF was only slightly more often preferred over WaveGAN.

The analysis of IS and FID leads to similar conclusions: TiFGAN-M showed a large improvement on both measures over the other conditions, with still a large gap to the real-data performance. On the other hand, comparing WaveGAN to TiFGAN-MTF, the results for both measures are mixed.

When evaluating the magnitude spectrograms generated by TiFGAN-M, TiFGAN-MTF, and those obtained from the real data, we notice that their consistencies are similarly close. Going a step further and applying PGHI to these magnitude spectrograms, the relative projection errors (RSPE) of the two networks are similar, but worse than those of the real signals, meaning that there is room for improvement in this regard. For the listening tests, PGHI was applied to the output of TiFGAN-MTF using the generated phase derivatives. In this setting, the RSPE was -7.5 dB, a substantially smaller value. This confirms our finding that phase reconstruction provides better results than phase generation by our network.

In summary, TiFGAN-M provided a substantial improvement over the previous state-of-the-art in unsupervised adversarial audio generation. Although the results for TiFGAN-MTF are not as clear, we believe that direct generation of phase could provide results on par or better than the magnitude alone and should be systematically investigated. Further research will focus on avoiding discrepancies between the phase derivatives and the log-magnitude.

5. Conclusions

In this contribution, we considered adversarial generation of a well understood time-frequency representation, namely the STFT. We proposed steps to overcome the difficulties that arise when generating audio in the short-time Fourier domain, taking inspiration from properties of the continuous STFT (Portnoff, 1976; Auger et al., 2012; Gröchenig, 2001) and from the recent progress in phaseless reconstruction (Průša et al., 2017). We provided guidelines for the choice of STFT parameters that ensure the reliability of phaseless reconstruction. Further, we introduced a new measure assessing the quality of a magnitude STFT, i.e., the consistency measure. It is computationally cheap and can be used to a-priori estimate the potential success of phaseless reconstruction. In the context of GANs, it can ease the assessment of convergence at training time.

Eventually, we demonstrated the value of our guidelines in the context of unsupervised audio synthesis with GANs. We introduced TiFGAN, a GAN directly generating invertible STFT representations. Our TiFGANs, trained on speech and music outperformed the state-of-the-art time-domain GAN both in terms of psychoacoustic and numeric evaluation, demonstrating the potential of TF representations in generative modeling.

In the future, further extensions of the proposed approach are planned towards TF representations on logarithmic and perceptual frequency scales (Brown, 1991; Brown & Puckette, 1992; Holighaus et al., 2013; 2019; Necciari et al., 2018).

Acknowledgments

This work has been supported by Austrian Science Fund (FWF) project MERLIN (Modern methods for the restoration of lost information in digital signals;I 3067-N30). We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research. We would like to thank the authors of (Engel et al., 2019) for providing us with details of their implementations prior to presenting it at ICLR 2019, allowing us to have their approach as a comparison. We would also like to thank the anonymous reviewers and Peter Balazs for their tremendously helpful comments and suggestions.

References

Alaifari, R. and Grohs, P. Phase retrieval in the general setting of continuous frames for banach spaces. *SIAM journal on mathematical analysis*, 49(3):1895–1911, 2017.

Alaifari, R. and Wellershoff, M. Stability estimates for phase retrieval from discrete gabor measurements. *arXiv preprint arXiv:1901.05296*, 2019.

Allen, J. Short term spectral analysis, synthesis, and modification by discrete fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 25(3):235–238, 1977. doi: 10.1109/TASSP.1977.1162950.

Arfib, D., Keiler, F., Zölzer, U., Verfaille, V., and Bonada, J. Time-frequency processing. *DAFX: Digital Audio Effects*, pp. 219–278, 2011.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *Proc. of ICML*, pp. 214–223, 2017.

Auger, F. and Flandrin, P. Improving the readability of time-frequency and time-scale representations by the reassignment method. *IEEE Trans. Signal Proc.*, 43(5): 1068 –1089,, may 1995.

Auger, F., Chassande-Mottin, É., and Flandrin, P. On phase-magnitude relationships in the short-time fourier transform. *IEEE Signal Process. Lett.*, 19(5):267–270, 2012.

Balan, R., Casazza, P., and Edidin, D. On signal reconstruction without phase. *Applied and Computational Harmonic Analysis*, 20(3):345–356, 2006.

Balazs, P., Bayer, D., Jaillet, F., and Søndergaard, P. The pole behavior of the phase derivative of the short-time fourier transform. *Applied and Computational Harmonic Analysis*, 40(3):610–621, 2016.

Bargmann, V. On a Hilbert space of analytic functions and an associated integral transform part i. *Communications on Pure and Applied Mathematics*, 14(3): 187–214, 1961. doi: 10.1002/cpa.3160140303. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160140303.

Brock, A., Donahue, J., and Simonyan, K. Large scale GAN training for high fidelity natural image synthesis. In *Proc. of ICLR*, 2019.

Brown, J. C. Calculation of a constant Q spectral transform. *The Journal of the Acoustical Society of America*, 89(1): 425–434, 1991.

Brown, J. C. and Puckette, M. S. An efficient algorithm for the calculation of a constant Q transform. *The Journal of the Acoustical Society of America*, 92(5):2698–2701, 1992.

Christensen, O. *An Introduction to Frames and Riesz Bases*. Applied and Numerical Harmonic Analysis. Birkhäuser Basel, Second edition, 2016. ISBN 978-3-319-25611-5; 978-3-319-25613-9.

Conway, J. B. *Functions of one complex variable*. Springer-Verlag New York [New York], 1973. ISBN 3540900624 0387900616 0387900624.

Dieleman, S. and Schrauwen, B. End-to-end learning for music audio. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 6964–6968. IEEE, 2014.

Dolson, M. The phase vocoder: A tutorial. *Computer Music Journal*, 10(4):14–27, 1986.

Donahue, C., McAuley, J., and Puckette, M. Adversarial audio synthesis. In *Proc. of ICLR*, 2019.

Engel, J., Resnick, C., Roberts, A., Dieleman, S., Norouzi, M., Eck, D., and Simonyan, K. Neural audio synthesis of musical notes with wavenet autoencoders. In *Proc. of ICML*, pp. 1068–1077, 2017.

Engel, J., Agrawal, K. K., Chen, S., Gulrajani, I., Donahue, C., and Roberts, A. GANSynth: Adversarial neural audio synthesis. In *Proc. of ICLR*, 2019.

Fan, Z.-C., Lai, Y.-L., and Jang, J.-S. R. SVSGAN: Singing voice separation via generative adversarial network. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 726–730. IEEE, 2018.

Faulhuber, M. and Steinerberger, S. Optimal gabor frame bounds for separable lattices and estimates for jacobi theta functions. *Journal of Mathematical Analysis and Applications*, 445(1):407–422, 2017.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Griffin, D. and Lim, J. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 32(2):236–243, 1984.

Gröchenig, K. *Foundations of Time-Frequency Analysis*. Appl. Numer. Harmon. Anal. Birkhäuser, 2001.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. Improved training of wasserstein GANs. In *Advances in Neural Information Processing Systems*, pp. 5767–5777, 2017.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pp. 6626–6637, 2017.

Holighaus, N., Dörfler, M., Velasco, G. A., and Grill, T. A framework for invertible, real-time constant-Q transforms. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(4):775–785, 2013.

Holighaus, N., Koliander, G., Průša, Z., and Abreu, L. D. Characterization of analytic wavelet transforms and a new phaseless reconstruction algorithm. *Preprint, submitted to IEEE Trans. Sig. Proc.*, 2019. URL `http://ltfat.github.io/notes/ltfatnote053.pdf`.

Huang, S., Li, Q., Anil, C., Bao, X., Oore, S., and Grosse, R. B. TimbreTron: A WaveNet (CycleGAN (CQT (Audio))) pipeline for musical timbre transfer. In *Proc. of ICLR*, 2019.

Janssen, A. From continuous to discrete Weyl-Heisenberg frames through sampling. *Journal of Fourier Analysis and Applications*, 3(5):583–596, 1997.

Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of GANs for improved quality, stability, and variation. In *Proc. of ICLR*, 2018.

Kingma, D. and Ba, J. Adam: A method for stochastic optimization. In *Proc. of ICLR*, 2015.

Le Roux, J., Kameoka, H., Ono, N., and Sagayama, S. Fast signal reconstruction from magnitude STFT spectrogram based on spectrogram consistency. In *Proc. Int. Conf. Digital Audio Effects*, volume 10, 2010.

Lyons, L. *A Practical Guide to Data Analysis for Physical Science Students*. Cambridge University Press, 1991.

Mallat, S. and Waldspurger, I. Phase retrieval for the cauchy wavelet transform. *Journal of Fourier Analysis and Applications*, 21(6):1251–1309, 2015.

Marafioti, A., Perraudin, N., Holighaus, N., and Majdak, P. A context encoder for audio inpainting. *Preprint, submitted to IEEE TASLP.*, 2018. URL `https://arxiv.org/pdf/1810.12138.pdf`.

Mehri, S., Kumar, K., Gulrajani, I., Kumar, R., Jain, S., Sotelo, J., Courville, A., and Bengio, Y. SampleRNN: An unconditional end-to-end neural audio generation model. In *Proc. of ICLR*, 2017.

Muth, J., Uhlich, S., Perraudin, N., Kemp, T., Cardinaux, F., and Mitsufuji, Y. Improving DNN-based music source separation using phase features. In *Joint Workshop on Machine Learning for Music at ICML, IJCAI/ECAI, and AAMAS*, 2018.

Necciari, T., Holighaus, N., Balazs, P., Průša, Z., Majdak, P., and Derrien, O. Audlet filter banks: A versatile analysis/synthesis framework using auditory frequency scales. *Applied Sciences*, 8(1:96), 2018.

Pascual, S., Bonafonte, A., and Serrà, J. SEGAN: Speech enhancement generative adversarial network. In *Proc. of Interspeech*, pp. 3642–3646, 2017.

Perraudin, N., Balazs, P., and Søndergaard, P. L. A fast Griffin-Lim algorithm. In *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2013 IEEE Workshop on*, pp. 1–4. IEEE, 2013.

Pons, J., Nieto, O., Prockup, M., Schmidt, E. M., Ehmann, A. F., and Serra, X. End-to-end learning for music audio tagging at scale. *CoRR*, abs/1711.02520, 2017. URL `http://arxiv.org/abs/1711.02520`.

Portnoff, M. Implementation of the digital phase vocoder using the fast fourier transform. *IEEE Trans. Acoust. Speech Signal Process.*, 24(3):243–248, 1976.

Průša, Z., Søndergaard, P. L., Holighaus, N., Wiesmeyr, C., and Balazs, P. The Large Time-Frequency Analysis Toolbox 2.0. In *Sound, Music, and Motion*, LNCS, pp. 419–442. Springer International Publishing, 2014. ISBN 978-3-319-12975-4. doi: 10.1007/978-3-319-12976-1_25.

Pruša, Z. STFT and DGT phase conventions and phase derivatives interpretation. Technical report, Acoustics Research Institute, Austrian Academy of Sciences, 2015.

Průša, Z., Balazs, P., and Søndergaard, P. A noniterative method for reconstruction of phase from STFT magnitude. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 25(5):1154–1164, 2017.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Proc. of ICLR*, 2016.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, pp. 2234–2242, 2016.

Shen, J., Pang, R., Weiss, R., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerry-Ryan, R., Saurous, R., Agiomyrgiannakis, Y., and Wu, Y. Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions. In *Proc. of ICASSP*, 2018.

Sotelo, J., Mehri, S., Kumar, K., Santos, J. F., Kastner, K., Courville, A., and Bengio, Y. Char2wav: End-to-end speech synthesis. In *Workshop on ICLR*, 2017.

Strohmer, T. Numerical algorithms for discrete Gabor expansions. In Feichtinger, H. G. and Strohmer, T. (eds.), *Gabor Analysis and Algorithms: Theory and Applications*, Appl. Numer. Harmon. Anal., pp. 267–294. Birkhäuser Boston, 1998.

Strohmer, T. and Beaver, S. Optimal OFDM system design for time-frequency dispersive channels. *IEEE Trans. Comm.*, 51(7):1111–1122, July 2003.

Van Den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. Wavenet: A generative model for raw audio. *CoRR abs/1609.03499*, 2016.

Van Den Oord, A., Li, Y., Babuschkin, I., Simonyan, K., Vinyals, O., Kavukcuoglu, K., van den Driessche, G., Lockhart, E., Cobo, L., Stimberg, F., Casagrande, N., Grewe, D., Noury, S., Dieleman, S., Elsen, E., Kalchbrenner, N., Zen, H., Graves, A., King, H., Walters, T., Belov, D., and Hassabis, D. Parallel WaveNet: Fast high-fidelity speech synthesis. In *Proc. of ICML*, pp. 3918–3926, 2018.

Warden, P. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209*, 2018.

Wexler, J. and Raz, S. Discrete gabor expansions. *Signal Processing*, 21(3):207 – 220, 1990. doi: https://doi.org/10.1016/0165-1684(90)90087-F.