A UNIFIED NEURAL ARCHITECTURE FOR INSTRUMENTAL AUDIO TASKS

Steven Spratley, Daniel Beck, and Trevor Cohn

School of Computing and Information Systems
The University of Melbourne, Victoria, Australia

ABSTRACT

Within Music Information Retrieval (MIR), prominent tasks — including pitch-tracking, source-separation, super-resolution, and synthesis — typically call for specialised methods, despite their similarities. Conditional Generative Adversarial Networks (cGANs) have been shown to be highly versatile in learning general image-to-image translations, but have not yet been adapted across MIR. In this work, we present an end-to-end supervisable architecture to perform all aforementioned audio tasks, consisting of a WaveNet synthesiser conditioned on the output of a jointly-trained cGAN spectrogram translator. In doing so, we demonstrate the potential of such flexible techniques to unify MIR tasks, promote efficient transfer learning, and converge research to the improvement of powerful, general methods. Finally, to the best of our knowledge, we present the first application of GANs to guided instrument synthesis.

Index Terms— music information retrieval, generative adversarial network, audio modelling, synthesis

1. INTRODUCTION

Within Music Information Retrieval (MIR), the prominent tasks of **pitch-tracking**, **source-separation**, **super-resolution**, and **synthesis** are usually treated as distinctly different tasks, employing widely varying techniques [1, 2]. We suspect some redundancy in this approach; humans can listen for particular instruments in an ensemble, track their behaviour, and imagine 'hearing' them without external stimuli, all as a by-product of learning each instrument's sound. Consequently, we expect that methods for processing audio should be adept in several such tasks, jointly in a multi-task learning setting.

Despite requiring different methods, time-frequency spectrograms are central to most current techniques. Recently, *WaveNets* [3] were shown to reconstruct such spectrograms more accurately than the long-standing Griffin-Lim algorithm [4], yet this has barely been applied outside of speech processing. Meanwhile, image translation models such as *Generative Adversarial Networks* (GANs) have achieved wide success; several papers have investigated their application to speech audio [5, 6]. Yet, little work has been done to investigate their general application across multiple tasks, or to instrumental audio. Therefore, to frame the aforementioned tasks as the same underlying problem is an attractive pursuit, potentially allowing research to leverage powerful new methods, efficiently train multi-task models, and converge on improving the one technique.

Music is sometimes conceived of as language without semantics; one cannot converse with music about *concepts*, yet it displays syntax and organisation at the levels of rhythm and harmony. Therefore, one ought to be able to consider its translation. As a sentence may be embellished or summarised, the same notes may be performed by an ensemble or a single instrument, differentiated by the number of frequencies superposed in the mix. Translation in this limited sense becomes the act of adding or stripping away frequencies (Figure 1). On one extreme, pitch-tracking can be thought of as distilling the fundamental frequency, F_0. Its opposite, synthesis, is therefore taking F_0 and introducing the rest of the spectrum. Source-separation of polyphonic audio becomes detecting the frequencies made by a single instrument and ignoring the rest, whereas super-resolution restores upper frequencies to low-fidelity recordings.

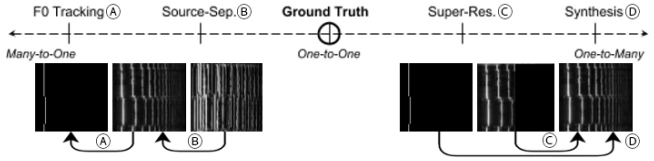

Fig. 1. Framing prominent instrumental audio tasks as translation. Illustrated with spectrograms; frequencies can be distilled in operations to the left, or added, to the right.

Given the proven utility of the time-frequency spectrogram in a wide range of audio tasks, the framing of these tasks as a translation problem, the emergence of image translation methods, and improved spectrogram reconstruction techniques, this paper seeks to develop a single model framework capable of each task, and their combination. Our pipeline consists of a conditional GAN that learns translations between mel-scaled spectrograms, which in turn, conditions a WaveNet synthesiser to reconstruct the final audio. This contributes a generalisable and end-to-end supervisable architecture that displays both competitive performance across a wide application, and further gains from joint multi-task learning.[1]

2. MODELS

2.1. Generative Adversarial Networks and pix2pix

Generative adversarial networks (GANs) [7] are frameworks consisting of generator and discriminator (G and D) subnetworks that learn via competition. $D(y;\theta_d)$ seeks to maximise the probability of correctly discerning whether y was taken from real data $x{\sim}p_{data}(x)$ or generated, counterfeit data. Simultaneously, $G(z;\theta_g)$ learns to map input noise $z{\sim}p_z(z)$ to x in order to produce increasingly convincing imitations. G is never shown the data it aims to generate, instead it relies solely on D's output as part of its own loss. The parameters θ_g are trained to minimise $\log\left(1 - D(G(z))\right)$, while parameters θ_d learn to maximise $\log D(x) + \log\left(1 - D(G(z))\right)$. Due to the codependence of the networks, the training scheme involves alternating their update phases. *pix2pix* (cGAN) [8] extends the framework by conditioning both subnetworks on images; G learns to map from noise to translation, conditioned on an observation, while D compares the same observation with a translation of unknown ori-

[1] Examples at `https://svenshade.github.io/GAN-WN/`.

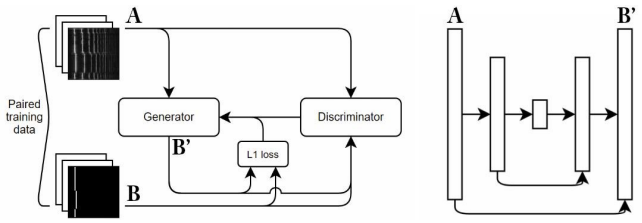

Fig. 2. Illustration of cGAN (left) and its U-Net generator (right).

gin (ground truth or imitation). This model also incorporates L1 distance to stabilise training via objective feedback. In pix2pix, D is a convolutional classifier which assumes independence between pixels of separate patches, allowing for fewer parameters and handling images of different sizes. G resembles a U-Net [9] autoencoder with skip connections between mirrored layers that circumvent the information bottleneck at the centre of the network by opening up other means for decoder layers to access low-level information (e.g. edges) that might be near identical with their paired encoder layers (see Figure 2). This feature is a strength of pix2pix, as there are many image translation tasks, including our own, that require some elements of the original composition to remain unchanged.

This work further demonstrates that loss functions needn't be explicitly specified by the researcher between tasks, as D effectively learns a bespoke loss. It therefore presents a flexible method that can be applied to a number of image translation problems, which our research aims to exploit in the spectrogram space.

2.2. WaveNet and Tacotron 2

A key hurdle presents itself when modelling raw audio. Capturing high-level image features (e.g. a cat) might require deeper neurons to be receptive to say, 200 pixels squared. A high-level audio feature (e.g. a bowed string) may be sustained over a second, increasing the receptive field requirement to tens of thousands of timesteps, even for a medium-fidelity signal. Although this is possible with large kernel sizes and strides, it doesn't present a way to overcome the second hurdle, which is to respect the sequential nature of audio. *WaveNets* [10] make use of causal, dilated convolutional layers. By both forcing connections to feed forward into later timesteps, and progressively dilating hidden layers, the resultant network can be visualised as a binary tree that slides along the data, enabling the receptive field to increase exponentially with layer depth. In this way, entire waveforms can be generated autoregressively; by classifying one timestep at a time given the existing series.

In a follow-up paper [3], *Tacotron 2* is described as a fully neural architecture for text-to-speech, making use of WaveNet as a synthesiser conditioned on features generated by the original Tacotron network, replacing the Griffin-Lim spectrogram-to-audio algorithm. Not only does this make the system supervisable end-to-end, it also allows compressed representations such as mel-spectrograms to be reconstructed directly, reducing network complexity.

3. THE GAN-WN ARCHITECTURE FOR VARIOUS INSTRUMENTAL AUDIO TASKS

3.1. Data and Representation

We chose to exclusively model the solo violin for three reasons. One, forming multiple 'deep' models for each task is infeasible with our compute resources. Two, the violin is notoriously hard to model due to its expressive range [11], serving as an effective proof-of-concept for many other instruments. Three, we expect it to facilitate efficient joint modelling of tasks. For pitch-tracking and synthesis tasks, we created paired multi-sine-wave (fundamental + first 5 harmonics) and violin audio tracks using chromatic scales, software instruments, and data from the *Bach10* dataset [12]. For super-resolution, we downsampled over 12 hours of live violin recordings [13, 14, 15, 16] in order to generate the paired low-fidelity track. Finally, for source-separation we used multi-track recordings from the *Bach10*, *Freischutz*, and *Phenicx-anechoic* datasets [12, 17, 18], and created our own synthetic multi-track data using MIDI files of Bach's *Four Orchestral Suites*[2], played through software instruments. We further tailored MIDI files by reducing the complexity of polyphonic phrases in order to clarify connection to F_0, and keeping instrument sources to under five. We chose to model at a literature-standard sample rate of 16 kHz, as the spectrum becomes increasingly sparse in upper frequencies captured by higher rates, meaning computationally-expensive diminishing returns. Once we had finalised our audio tracks, they were compressed and normalised, segmented into chunks, and processed via the short-time Fourier transform (STFT) [4]. STFT parameters (hop size of 49 samples, and window size of 1024 and length of 640 samples) were co-determined with chunk duration (1550ms), to settle on a representation that displayed clear frequency lines when viewed as an image at the target resolution of 256x256 pixels. Finally, we mel-scaled all data to efficiently model for human frequency perception, and reclaim memory for larger kernels and layers.

3.2. Translation

Our method extends the pix2pix method presented in [8] (using the official public implementation) in order to fit a translation model to each of our datasets of paired spectrograms. In each case, once the generator, G is properly trained, testing becomes a matter of converting audio of arbitrary length to its mel-spectrogram representation and applying G convolutionally. In this domain, pitch-tracking can be thought of as semantic segmentation in the same way that a satellite image might be translated into a road map. Source-separation becomes a denoising problem, relaying patterns related to the signal and ignoring others. Super-resolution is analogous to inpainting, where the first half of the spectrogram — the low frequencies — is used to fill in the blank half. Finally, synthesis becomes a style transfer problem, where the model has extracted the overall harmonic 'style' of an instrument, and seeks to apply this to sine-waves serving as a harmonic *blueprint*. After a preliminary grid search over hyperparameters, we adopted a kernel size of 8x8 for both G and D networks and optimised for least-squares loss.

3.3. Reconstruction

We present three spectrogram reconstruction methods. The first involves naively rescaling frequencies before processing with Griffin-Lim, which displays noticeable artefacts and a lack of detail in higher frequencies. The second method improves on this by inserting a secondary cGAN process after rescaling, trained on ground-truth and mel-scaled-rescaled data, to further restore the linear spectrograms. The third method makes use of the WaveNet architecture to circumvent both lossy rescaling and Griffin-Lim reconstruction, locally conditioned on spectrograms to guide generation. We trained one WaveNet model on the task that had the most available data (super-resolution). By partitioning into equal test and train ($\sim$6 hours of au-

[2] `http://www.jsbach.net/midi/`

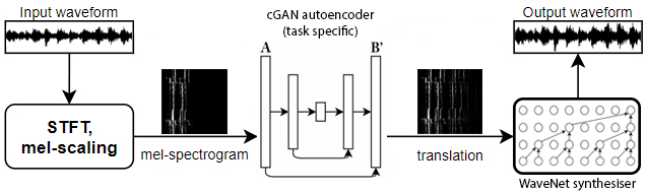

Fig. 3. The GAN-WN architecture.

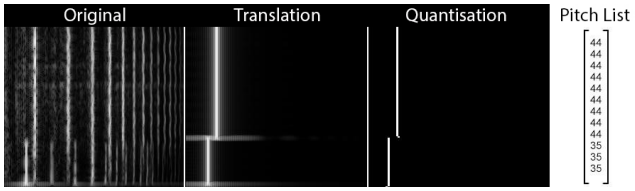

Fig. 4. The pitch quantisation process, proceeding left to right.

Method	Correct	Total	Precision	Mean error
Ableton	109 / 130	157	69.43%	32.69
cGAN	**119 / 130**	**122**	**97.54%**	**17.35**

Table 1. Pitch-tracking results, including correct/total notes found.

dio each), our cGAN produced 6 hours-worth of spectrograms from unseen data, which were then paired with their ground truth audio in order to train WaveNet. In doing so, we train GAN-WN as a cascade architecture, making the reconstruction stage more robust to translation errors. We illustrate the GAN-WN architecture in Figure 3. Memory resources for WaveNet limited training instances to $\sim$30k steps, or 1.88s in duration, which perfectly suited the 1.55s chunk size represented by our datasets. Noticing that our first WaveNet over-emphasised harmonics, we implemented a technique to further guide audio called *teacher-weighted generation*, which makes use of the original, low-fidelity signal being restored, by first interpolating it to the target sample rate, and taking a weighted average between each timestep generated by the network and the corresponding interpolated timestep, by a set factor. That prediction, $f^{-1}((f-1)x_i + y_i)$, where x_i is the timestep from the interpolated signal, y_i is WaveNet's current prediction, and f is our weight factor, is fed back into the network to discourage errors from being propagated. We found that a factor of 20 was sufficient to generate audio that was less prone to squealing, at the cost of some detail.

4. EXPERIMENTS AND RESULTS

After fitting a dedicated cGAN model to each of the four tasks' datasets, we evaluate on the bases of spectrogram distance as well as human audition. For measuring the former, we report both percentage difference (normalised L1 distance) as well as structural similarity (SSIM), which aims to independently consider structural information from illumination [19]. For the latter, we hosted an online, APE-style audition test [20] for twenty students and colleagues from the authors' personal networks. This evaluation was designed to facilitate comparison between multiple, time-aligned audio clips, by presenting the participant with a series of 20 horizontal scales, featuring sets of 4-6 audio clips corresponding to the same 10-second test instance as processed by the methods. Clip positions and labels were randomised at test time to avoid bias. Participants were tasked with auditioning each set and assessing each clip against a continuous Likert scale, across *bad*, *poor*, *fair*, *good*, and *excellent* markers. This format ensured that for each instance, all of the methods were compared with each other, and with finer resolution. The test was built using the *Web Audio Evaluation Tool* (WAET) [21].

4.1. Harmonic Distillation: Pitch-Tracking and Source-Separation

The ubiquitous baseline for pitch-tracking, *pYIN* [22], performed poorly over our data due to several octave errors. This prompted us to use the more sophisticated audio-to-MIDI function in the commercial digital audio workstation, *Ableton Live 9*. Unlike linearly-scaled spectrograms, training the model on the mel-scale ensures that the columns (filterbanks) closely correlate with musical pitch; understanding cGAN's pitch predictions was a simple matter of recording the filterbank number of the brightest pixel in each row (Figure 4). Over our test set, we tallied correctly predicted notes from both our method and the baseline, but to credit notes that were mistimed yet otherwise pitched properly, and investigate alignment of note onsets and offsets, we also tallied incorrectly classified timesteps (pixel rows), which we report as mean error per spectrogram (Table 1). Each timestep was considered incorrect if it was neither in, nor neighbouring, the correct row.

For source-separation, we used both blind and supervised baselines; the non-negative matrix factorisation (NMF) method implemented in the *Flexible Audio Source Separation Toolbox* [23], and the convolutional neural network-based method (CNN) detailed in [24]. The latter technique is chosen due to the high similarity of training data it shares with our method (Bach10 and pre-recorded violin samples), providing a more challenging baseline. The test piece — a Bach piano and violin duet — was chosen to better test how the methods fare on real-world data; neither the piano, music, nor number of sources, is featured in the training data for either method. See Table 2 for spectrogram results. For the listening tests in Figure 5, the prompt: "Overall Success: Removing Piano, Keeping Violin" was given to participants to rate against.

4.2. Harmonic Addition: Super-Resolution and Synthesis

For super-resolution, as in [25] we used interpolative baselines (linear, cubic b-spline) [26], which were compared against our pipeline using all three reconstruction methods (Table 2). While [25] itself provides a recent, informed technique, interpolation is sufficient for proof-of-concept, keeping to our aim of investigating the generality of our method, and mitigating the cost of training another baseline. We tested over solo violin audio from a different recording environment, downsampled to 4 kHz in order to perform 4x upsampling back to 16 kHz. We include the outcomes of the two super-resolution listening tests in Figure 5; participants were given the prompt "Overall Quality". While pixel-wise loss is a useful metric for distillation tasks, it becomes decreasingly useful the more we introduce new frequencies, where the real aim is to fill in sound that is believable, instead of meeting ground truth. We chose to evaluate synthesis by ablation; we remove D and train a baseline autoencoder on L1 distance alone. We train both methods over the synthesis dataset, this

	Source-Separation			*Super-Resolution*		
	cGAN	**NMF**	**CNN**	**cGAN**	**Cubic**	**Linear**
% error	**4.21**	4.96	5.24	**1.65**	2.42	2.63
SSIM	**0.52**	0.50	0.47	**0.68**	0.53	0.51

Table 2. Spectrogram performance over source-separation and super-resolution test sets.

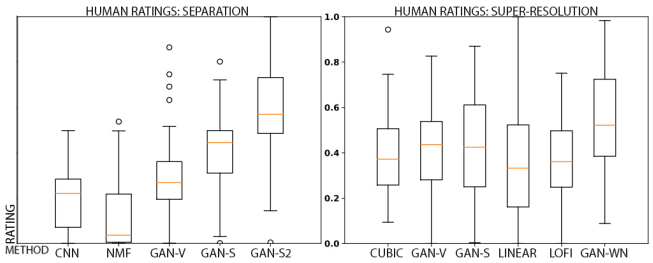

Fig. 5. Listening test plots. GAN-V refers to naive reconstruction, while GAN-S & S2 use 1 & 2 secondary cGAN stages respectively.

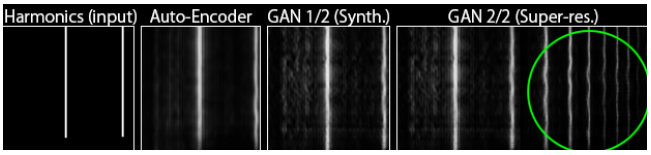

Fig. 6. Comparing synthesis models. Note the realistic trailing vibrato waves reproduced (encircled).

time making use of the aforementioned multi-sine-wave (harmonics) track; the single sine-wave track stalled training due to its sparsity. We also use our synthesis model in conjunction with the super-resolution model to avoid redundancy; we train the former on the simpler task of translating harmonics to 4 kHz violin audio, and then leverage the latter, trained on far more data, to inpaint the rest (Figure 6). During training, the capabilities of each model became clear, with the autoencoder requiring three times as many epochs as the cGAN model before reproducing clear noise + harmonics patterns.

4.3. Jointly Learning to Track, Separate, and Enhance

As stated earlier, there ought to be potential for significant transferable learning, with the acquisition and refinement of an instrument's audio profile facilitating multiple tasks. To pursue this, we formed a combined dataset of equal parts of our original sets, and trained a joint model by increasing the input channels of our cGAN to three (we collapsed synthesis and super-resolution to the one task, *enhancement*). We chose to compare this model against the dedicated source-separation model, given its complexity. We believe our results encourage such an approach (Figure 7), confirming our expectation that kernels not only ought to be shared efficiently between tasks, but also cross-benefit, as shown by the increased performance our joint model achieved, in fewer iterations, despite only processing the same amount of task-specific data as the dedicated model.

4.4. Discussion

We observe promising results; the models outperformed baselines, and subjective tests displayed clear preference for our method. For pitch-tracking, a common error made by *Ableton* was incorrectly timed note onsets (difficult for continuous-excitation instruments); in its defence, our method was trained to handle solo violin audio, and the metric we used only registered one correct note per ground truth note, which penalised Ableton's subdivided notes. Across distillation tasks, cGAN showed greater precision in the frequencies it detected, sometimes trading off detail in the high frequencies. Further tuning of our loss function could help approach CNN's accuracy in that regard, although CNN underperformed overall, conceivably due to it being trained to separate a set four sources. We also tested Signal-to-Distortion and Source-to-Interference ratios (SDR, SIR) for source-separation, in order to better quantify

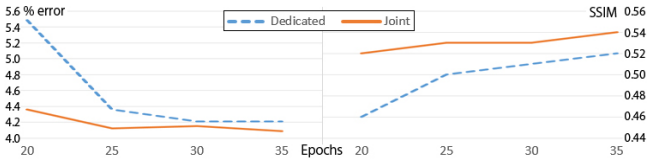

Fig. 7. Comparing joint vs. dedicated models.

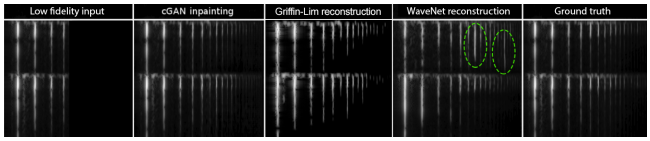

Fig. 8. Examples from our reconstructive processes on super-resolution. Whistling artifacts (encircled) introduced by WaveNet, and note the thresholding degradation from descaling & Griffin-Lim.

performance. However, we noticed a severe drop in SIR (from 25dB, down to 16dB) after our second cGAN restoration, as well as non-competitive SDR scores. Curiously, this is not consistent with the results of the subjective listening test, and we hypothesise that such success is due to the generative nature of the GAN process; it is liable to fill in statistically-consistent detail, even if such detail is not present in the signal. This marked difference in the way our technique generates audio compared to baselines makes it tricky for objective comparison, and we encourage future research in perceptual metrics for GAN-based MIR systems. This finding reveals a weakness in our technique as compared to specialised approaches, as source-separation arguably has a ground-truth that needs to be uncovered. To re-state; our approach is not to overtake state-of-the-art, but to explore and give credence to the efficient research strategy of investing in general techniques that model with less redundancy in training, since any data might increase performance in all tasks.

For addition tasks, sparsity issues (mapping too few frequencies to complex spectra) were aided by mel-scaling. WaveNet was able to model noise components better than Griffin-Lim, yet was let down by artefacts; we hypothesise this is analogous to positive feedback in language models, where a particular phrase predictably sparks others, propagating error. This was alleviated by teacher-weighted generation, at the cost of some detail in the upper spectrum. See Figure 8 for a comparison of spectrograms output by the different processes. Regarding algorithm performance, cGAN translation was more than an order of magnitude faster than the reconstruction stage, at ∼19x realtime. Our implementation of WaveNet[3] was slower, only able to produce ∼50 raw audio samples a second. Griffin-Lim reconstruction was much faster (∼0.78x realtime), allowing ∼0.5x realtime speed to complete any task using the cGAN-S method overall, and a ∼4x speedup compared to source-separation using our NMF baseline. All models were trained using an NVIDIA GTX 1080ti GPU.

5. CONCLUSIONS AND FUTURE DIRECTIONS

In this work, we have demonstrated how recent innovations in speech and image processing can impact diverse MIR problems, contributing to their unification in theory and practice. Our approach, GAN-WN, is a general, supervisable method that can be jointly-trained, retaining competitive performance across most tasks. By making use of STFT phase data, lossy spectrogram transformations might become redundant in future iterations, reducing artefacts. Finally, adding expressive control of synthesis, and pursuing multi-instrument modelling, are both promising extensions.

[3] Based largely on `github.com/r9y9/wavenet_vocoder`.

6. REFERENCES

[1] Emilia Gmez, Anssi Klapuri, and Benot Meudic, "Melody description and extraction in the context of music content processing," *Journal of New Music Research*, vol. 32, no. 1, pp. 23–40, 2003.

[2] Dominic Ward, Russell D Mason, Ryan Chungeun Kim, Fabian-Robert Stöter, Antoine Liutkus, and Mark D Plumbley, "Sisec 2018: state of the art in musical audio source separation-subjective selection of the best algorithm," in *Proceedings of the 4th Workshop on Intelligent Music Production, Huddersfield, UK, 14 September 2018*. University of Huddersfield, 2018.

[3] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, and RJ Skerry-Ryan, "Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions," *arXiv preprint arXiv:1712.05884*, 2017.

[4] Daniel Griffin and Jae Lim, "Signal estimation from modified short-time fourier transform," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1984, vol. 32, pp. 236–243.

[5] Santiago Pascual, Antonio Bonafonte, and Joan Serr, "SEGAN: Speech enhancement generative adversarial network," in *Proc. Interspeech 2017*, 2017, pp. 3642–3646.

[6] Yang Gao, Rita Singh, and Bhiksha Raj, "Voice impersonation using generative adversarial networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2506–2510.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[8] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1125–1134.

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, 2015, pp. 234–241.

[10] Aäron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio.," in *SSW*, 2016, p. 125.

[11] Graham Keith Percival, *Physical modelling meets machine learning: performing music with a virtual string ensemble*, Thesis, 2013.

[12] Zhiyao Duan, Bryan Pardo, and Changshui Zhang, "Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions," in *IEEE Transactions on Audio, Speech, and Language Processing*, 2010, vol. 18, pp. 2121–2133.

[13] "G.P. Telemann's Twelve Fantasias for Solo Violin [CD]," Harmonia Mundi, 1996.

[14] "Tartini Solo Violin [CD]," Harmonia Mundi, 1998.

[15] "Bach's Sonatas and Partitas for Solo Violin [CD]," Deutsche Grammophon, 1998.

[16] "The Westhoff Suites for Solo Violin [CD]," Capriccio, 2004.

[17] Thomas Prätzlich, Meinard Müller, Benjamin W Bohl, Joachim Veit, and Musikwissenschaftliches Seminar, "Freischütz digital: demos of audio-related contributions," in *Demos and Late Breaking News of the International Society for Music Information Retrieval Conference (ISMIR), Málaga, Spain*, 2015.

[18] Marius Miron, Julio J Carabias-Orti, Juan J Bosch, Emilia Gmez, and Jordi Janer, "Score-informed source separation for multichannel orchestral recordings," *Journal of Electrical and Computer Engineering*, vol. 2016, 2016.

[19] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[20] Brecht De Man and Joshua D Reiss, "APE: Audio perceptual evaluation toolbox for MATLAB," in *Audio Engineering Society Convention 136*. Audio Engineering Society, 2014.

[21] Nicholas Jillings, David Moffat, Brecht De Man, and Joshua D. Reiss, "Web Audio Evaluation Tool: A browser-based listening test environment," in *12th Sound and Music Computing Conference*, July 2015.

[22] Matthias Mauch and Simon Dixon, "pYIN: A fundamental frequency estimator using probabilistic threshold distributions," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 659–663.

[23] Yann Salaün, Emmanuel Vincent, Nancy Bertin, Nathan Souviraà-Labastie, Xabier Jaureguiberry, Dung T Tran, and Frédéric Bimbot, "The flexible audio source separation toolbox version 2.0," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014.

[24] Pritish Chandna, Marius Miron, Jordi Janer, and Emilia Gómez, "Monoaural audio source separation using deep convolutional neural networks," in *International Conference on Latent Variable Analysis and Signal Separation*, 2017, pp. 258–266.

[25] Volodymyr Kuleshov, S. Zayd Enam, and Stefano Ermon, "Audio super resolution using neural networks," *CoRR*, vol. abs/1708.00853, 2017.

[26] Hsieh Hou and H Andrews, "Cubic splines for image interpolation and digital filtering," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1978, vol. 26, pp. 508–517.