Timbre Transfer with Variational Auto Encoding and Cycle-Consistent Adversarial Networks

Russell Sammut Bonnici
School of Electronic Engineering and Computer Science
Queen Mary University of London
United Kingdom, E14NS
ec20074@qmul.ac.uk

Martin Benning
School of Mathematical Sciences
Queen Mary University of London
United Kingdom, E14NS
m.benning@qmul.ac.uk

Charalampos Saitis
School of Electronic Engineering and Computer Science
Queen Mary University of London
United Kingdom, E14NS
c.saitis@qmul.ac.uk

Abstract

This research project investigates the application of deep learning to timbre transfer, where the timbre of source audio can be converted to the timbre of target audio with minimal loss in quality. The adopted approach combines Variational Autoencoders with Generative Adversarial Networks to construct meaningful representations of source audio and produce realistic generations of target audio and is applied to the Flickr 8k Audio dataset for transferring the vocal timbre between speakers and the URMP dataset for transferring the musical timbre between instruments. Furthermore, variations of the adopted approach are trained, and generalised performance is compared using the metrics SSIM (Structural Similarity Index) and FAD (Frechét Audio Distance). It was found that a many-to-many approach supersedes a one-to-one approach in terms of reconstructive capabilities, while one-to-one showed better results in terms of adversarial translation. The adoption of a basic over a bottleneck residual block design is more suitable for enriching content information about a latent space. It was also found that the decision on whether cyclic loss takes on a variational autoencoder or vanilla autoencoder approach does not have a significant impact on reconstructive and adversarial translation aspects of the model.

Acknowledgments

The research work disclosed in this publication is funded by the ENDEAVOUR Scholarships Scheme (Malta). The scholarship is part-financed by the European Union – European Social Fund (ESF) under Operational Programme II – Cohesion Policy 2014-2020, "Investing in human capital to create more opportunities and promote the well-being of society". We would like to express our gratitude to Ehab Albadawy for sharing his source code, feedback, and giving advice that helped resolve a number of conceptual confusions. We would also like to thank Ben Hayes and Cyrus Vahidi from Queen Mary University of London for providing us with insights on works in this problem domain.

Preprint. Under review.

1 Introduction

Timbre transfer is a task concerned with modifying audio samples such that their timbre is reformed while their semantic content is persisted. Through this, utterances of a speaker (referred to as the source) can be changed such that they sound like they were spoken by another speaker (referred to as the target). Recordings of a source instrument can be manipulated in a similar way such that they sound like another target instrument played them. Applications of effective timbre transfer would benefit areas such as voice anonymisation, music production, and data augmentation. The challenge in making the modification take place first lies in how exactly timbre can be captured.

Timbre is formally defined as the quality of an audio stimulus in which a listener can distinguish two sounds with a factor separate from loudness and pitch [1]. As reflected by how its definition describes what it is not, timbre is highly abstract and hard to determine concisely. Despite sometimes getting referred to as tone colour, it is harder to quantify than visual colour. Visual colour is commonly defined in three dimensions with an RGB model, and though previous research has determined a three-dimensional model for timbre [2], it is still not as clear cut.

Explicit characteristics such as spectral envelope and time envelope help determine timbre for instruments, but there are still implicit characteristics that contribute to painting the complete picture. Also, musicians with more exposure to instruments of varying timbre are better at identifying them [3], indicating a direct proportionality between exposure and timbral understanding. These points motivate the use of deep learning, where from data, hard-to-define patterns of timbre can be learnt by models non-linearly, and such models can be applied for related discriminative and generative tasks.

Generative modelling is a task that has been increasingly getting more attention in recent years. Like discriminative modelling, intrinsic patterns about a collection of samples are learnt. Unlike discriminative modelling, it does not deduce conclusive information about the samples but uses the learnt patterns to generate new samples for a target sample distribution. From the research field of computer vision, a variety of performant generative models have been proposed for tasks such as data generation and style transfer. Most recent models extend from Variational Autoencoders [4] and Generative Adversarial Networks [5].

Generative Adversarial Networks (GANs) are an approach to generative modelling that aim to achieve realistic results with a discriminative overseer. GANs consist of two networks referred to as a generator and discriminator. The generator is concerned with generating sufficiently realistic data such that when it is compared to target data by the discriminator, it is indistinguishable. Meanwhile, Variational Autoencoders (VAEs [4]) take on a reconstructive approach. VAEs are split into two networks referred to as the encoder and decoder. Together, they compose a unified hourglass-like architecture. The encoder learns to compress input data into a highly abstract latent space at a bottleneck central to the model. Since the autoencoder is variational, it learns to model latent data such that it resembles a specified data distribution (typically Gaussian) for a better consistency in description at the bottleneck. For this, Kullback Leibler divergence is used to estimate the log difference between the probability of data in the predicted distribution and the probability of data in the desired distribution. Lastly, the decoder learns to decompress latent data from the bottleneck to the dimensional space of input data.

In training, VAEs are more stable than GANs. Since VAEs have weights tuned with respect to loss computations of their own output, their optimal state lies within a local minimum. Like most deep learning models, the loss minimises asymptotically. Meanwhile, GANs tune the generative component based on loss computations from a separate adversarial discriminator network. This makes their loss function non-linear, making it more challenging for an optimiser to works out the optimal state of loss.

The instability of GANs makes them sensitive to data and design decisions, as well as susceptible to mode collapse. This is a typical failure case where the generator learns to "cheat" the discriminator by mapping more than one input sample to the same output. Despite their notorious instability, when configured right, GANs can achieve significantly realistic results. Their vulnerability to mode collapse suggests that they would benefit from architectural decisions in the generator that better capture the latent meaning of input samples. An approach to this would be to remodel the generator into a VAE, yielding a VAE-GAN model design. This approach is the focal point of this work where by combining the two generative techniques, the adversarial aspect becomes more stabilised, while the reconstructive aspect benefits from more motivation for realism.

2 Related Work

2.1 Image Style Transfer

UNIT [6] is a notable model that motivates the adoption of a VAE-GAN approach for style transfer. It focuses on transferring styles of a source image to that of a target image. For a pair of style domains, they train the transfer in both directions. For each transfer direction, they use an encoder-decoder-discriminator pathway. The encoder-decoder section follows the reconstructive objective of VAEs (motivating content persistence). Meanwhile, the whole path with the discriminator included follows the adversarial objective of GANs (motivating style transfer). At the bottleneck, they enforce a shared latent space by computing the VAE objective in both directions. They also share weights of the last layer of the encoders and share the weights of the first layer of the decoders for motivating a shared latent space.

The reconstructive component of the VAE objective in UNIT was computed using an error criterion L1, which computes the total difference between the absolute magnitudes of a reconstructed image and its original version. Similarly, CycleGAN [7] trains in both directions and uses L1 for comparing the quality of a reconstruction to the original input. Both models are cyclically consistent since the cyclic L1 loss acts as a prior that allows the applicability of style transfer to unpaired data, where the content between two images is different. The main difference is that CycleGAN does not assign a VAE objective to the generator section, and so in the cyclic loss, there is no inclusion of Kullback–Leibler divergence alongside the L1 reconstructive component. As a result, CycleGAN learns how to transfer styles from a lower level.

The shared latent space of UNIT aims to address style transfer from a probabilistic modelling perspective. They reason that the goal is to capture the joint distribution of two style domains in order to transfer between them. It is tough to do this when data is unpaired and not captured at a high enough level. The shared latent space aims to better capture the joint distribution by emphasising source content and deemphasising source style. Intuitively, it represents content with less individuality at the bottleneck so that it becomes easier for the decoder to introduce target style to it.

2.2 Timbre Transfer

A number of works concerned with audio style transfer burrow intuition from image style transfer models. Timbre transfer is a subset of audio style transfer, for which timbre is taken as the style of interest in the audio domain. Here, there are typically two different design paradigms that researchers follow. They either follow a time-domain approach, where an end-to-end deep learning model deals with audio directly and at a low level. Alternatively, they may follow a time-frequency procedure, where audio is handled more indirectly but at a higher level. In this approach, the data is further processed for less complexity, with two deep learning models used for a high-quality transfer. The first model is concerned with performing style transfer on spectrogram representations of the audio, and the second model is concerned with vocoding the results of generated spectrograms back to realistic audio.

AutoVC [8] deals with utterances of speakers in the time-frequency domain and proposes a style transfer model that follows a vanilla autoencoder architecture. Like UNIT, they motivate content persistence in the bottleneck and style adaptation in the decoder but achieve it without an adversarial component. Their model consists of two encoders; a content encoder and a style encoder. The content encoder focuses on embedding the content of source utterances, whereas the style encoder focuses on embedding the style of target utterances. The decoder then accepts both content and style codes as input so that its transferred output is the amalgamation of source content and target style. The purpose of the style encoder replaces the purpose of the adversarial component in UNIT for target motivation. Furthermore, they focus on mel-spectrogram representations and use WaveNet [9] to convert generated mel-spectrograms to audio.

TimbreTron [10] focuses on recordings of instruments in the time-frequency domain. They also use WaveNet for vocoding and follow the approach of CycleGAN for their style transfer model. Initially, they perform the style transfer on vanilla spectrograms computed using a Short Time Fourier Transform (STFT) but report issues with correctly transferring low pitches as well as having output pitches generally randomly vary to a degree. To overcome these issues, they apply the style transfer to CQT spectrogram representations instead, where a higher resolution for lower frequencies is captured

and pitch equivariance is maintained. Though an improvement, their generated results still lack in quality. Their change of focus to CQT spectrograms likely tries to make up for the lack of latent representation capabilities in CycleGAN. Rather than adopting a different spectrogram representation, it may be better to adopt a VAE-GAN approach such as UNIT that still involves cyclic consistency but can also embed content at a higher level.

Mor et al [11] proposed a model in the time domain for universal music translation. Their style of interest involves timbre but also extends to a broader aspect, inclusive of composition style. They use a denoising autoencoder architecture, where input data is augmented and learnt to be reconstructed with respect to non-augmented data. The encoding and decoding components are designed following the principles of WaveNet since it excels at generation in the time domain. For the augmentation, pitch is randomly varied within a deviation of 0.5 half-steps. They motivate this so as to capture content representations at a higher level in the bottleneck. Though it yields an improvement in content generalisation, the augmentation may contribute to off-key pitch in the output, which is especially undesirable in musical contexts. With respect to the bottleneck, they apply an efficient shared latent space constraint by using the same encoder for all source domains. Motivating the generalisability of a universal encoder, they achieve applicability to source domains that were unseen from training. They also use a domain confusion loss to discourage the inclusion of style elements in the content embeddings.

AlBadawy and Lyu [12] proposed a model for voice conversion in the time-frequency domain. Their style transfer model serves as an extension of UNIT, where a shared latent space is encouraged with a universal encoder (much like Mor et al). They also introduce a latent loss to penalise significantly different averages of latent codes originating from other style domains (also similar to the purpose of the domain confusion loss in Mor et al). It encourages embeddings to be more independent of style by making content distributions align closer in the bottleneck. Their adaptation of Mor et al's intuition to the time-frequency domain eases training expenses as well as architectural complexity. A more semantically rich content embedding may also be achieved with the variational autoencoding elements of UNIT. Much like other related works, they apply their style transfer to mel-spectrograms and use WaveNet for vocoding back into audio.

This project adopts the VAE-GAN approach taken by AlBadawy and Lyu as their work demonstrates highly realistic results for the voice conversion between two speakers. Investigating potential model improvements as well as extending its applicability to the context of instruments poses an interesting study case. Like speech conversion, timbre remains the main style of interest, but for instruments, the content of interest becomes melodic sequences rather than linguistic sentences. Their time-frequency approach also makes the training and comparing of experiments more feasible both in terms of time and resource requirements. By decoupling the audio generation process (spectrogram vocoder) from the style transfer process, different style transfer experiments may be trained without having to retrain the audio generation process each time. Also, this design is not limited to a specific approach for audio generation, making it easily applicable to a variety of vocoders.

3 Method

3.1 Data Preprocessing

Prior to computing mel-spectrograms for the style transfer model, the input audio is preprocessed in a number of steps. Firstly, the audio is ensured to match a specified sample rate of 16,000 Hz. If not, then the audio is resampled accordingly. The volume is then subtly equalised with root mean square normalisation, where if the root mean square amplitude of the audio is lower than a target amplitude of -30dB, then the audio is normalised to that amplitude. Lastly, long silences from the audio are masked out to remove background noise at segments where no sound events are present. Note that the low amplitude of -30dB was chosen arbitrarily, subtly equalising low segments of the audio as a result. Higher amplitudes such as -10db may be set in future work. For each of the processed audio samples, mel-spectrograms are then computed with 128 mel frequency bins, a hop size of 200 samples, and a Hanning window of size 800 samples for each frame. Also, magnitude values for each mel-spectrogram are logarithmically scaled and normalised with min-max normalisation for a faster convergence in training. For each audio source (whether speaker or instrument), whole audio

and corresponding mel-spectrogram samples are organised into subsets of 80% for training, 10% for validation, and 10% for testing.

3.2 Components and Data Input

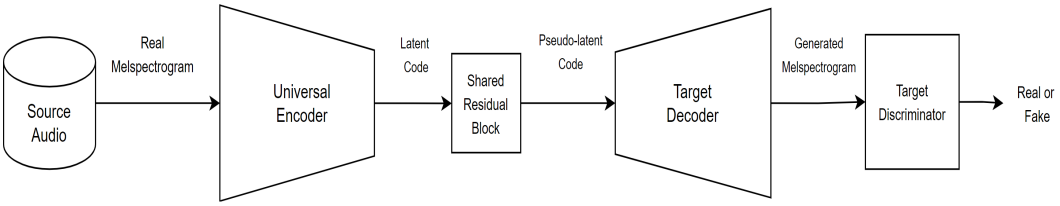


Figure 1: The design of a single path in the implemented VAE-GAN model, where the transfer of one timbre to another is learnt in a mel-spectrogram format. In training, two paths allow a one-to-one style transfer between timbres in both directions. For each transfer direction, the encoder aims to extract the content of the source audio regardless of timbre, whereas the decoder aims to introduce the timbre of the target audio to the content as motivated by the discriminator.

The style transfer model involves one encoder, a shared residual block for decoding at a shallow level, and multiple decoder-discriminator pairs specific to target domains. An illustration of a single path within this model is presented in Figure 1. Only two paths are utilised for a one-to-one style transfer in the initial implementation, where for a forward path that converts timbre A to timbre B, there also exists an inverse path that converts B to A. This is necessary for cyclic consistency. For each source, only one decoder-discriminator pair is trained for a target. This design may be extended to many-to-many style transfer by having each source simultaneously train different decoder-discriminator pairs for multiple targets. Not only would this make it easier to extend the model to numerous examples of timbre transfer, but the shared encoder would further benefit in terms of generalisation.

The input mel-spectrograms are subsamples of the full mel-spectrograms such that their width corresponds to a length of 128 frames (1.6 seconds), making the input mel-spectrograms of a resolution size of 128x128. This was necessary to have a standardised input size since the model largely depends on convolutional operations. For each of the domains, a combination of samples was randomly selected from the training set. From each sample, a 128 frame excerpt is extracted between random frame indices. This achieves a random data selection across domains and in terms of time localisation within each mel-spectrogram.

3.3 Architecture

The architectural details of the model's components are primarily congruent to the design choices initially proposed by UNIT [6]. The shared encoder involves two principal stages; a convolutional stage and a residual stage. It firstly pads input with a 3x3 reflection padding. After which, padded information is passed through 3 convolutional blocks that each downsample feature maps by a factor of 2. The first input channel is set to 1 to correspond to the loudness dimension of the mel-spectrogram. The 3 convolutional blocks then have output channels set at 32, 64, and 128 consecutively such that the feature map dimensions are stretched in-depth and compressed in width and height (leading to a compressed latent space). Each convolutional layer is followed by a LeakyReLu activation with an 0.2 negative slope and followed by an instance normalisation layer. The first convolutional layer has a kernel size of 7x7 whereas the following two have 4x4 kernel sizes. Following the convolutional blocks are 3 residual blocks.

The residual blocks involve skip connections that enrich the representation of data leading to the bottleneck with residual priors. The skip connections are implemented such that for each block, the input feature map is connected to the output feature map via a summation operation. The initially implemented blocks correspond to the basic residual block design proposed by ResNet [13], where each block consists of two convolutional layers of a 3x3 kernel size and only the first layer has ReLu applied after it. Unlike the ResNet design, here input is padded with a reflection padding of 1x1 and instance normalisation is applied after the convolutional layers instead of batch normalisation.

Instead of using basic residual blocks, bottleneck residual blocks (also proposed by ResNet) may be considered. Bottleneck residual blocks contain 3 convolutional layers of kernel sizes 1x1, 3x3, and

1x1. Due to the compression that their 3x3 convolutional layer provides, feature maps are reduced in size and so the operations required by a forward pass are reduced. This would benefit the model in terms of easing training expenses, but it may also make training more susceptible to vanishing gradients since the feature maps around the main bottleneck are already at a sufficiently compressed depth.

The output of the last encoder residual block is taken as a latent mean μ. The reparameterisation trick is then ensued where a random sample is selected from a zero-mean Gaussian distribution with a unit variance of 1. The sum of the random sample and the latent mean is computed to achieve a latent code z, which aims to represent the content of the input mel-spectrogram at the bottleneck.

The latent code z is taken as input to the decoder. The decoder follows the exact same architecture as the universal encoder but in a mirrored manner so that a 128x128 mel-spectrogram can be reconstructed as output. The 3 residual blocks follow the same setup as in the encoder, without much need for a reordering since each block has its output dimensions correspond to their input dimensions. The only difference is that instead of all residual blocks being shared across different source domains, only the first one is. The other two are specific to the target domain. This is meant to help motivate content abstraction, where content is decoded once in a similar manner before getting decoded further with respect to the target. The following 3 convolution blocks are also specific to the target and utilise transposed convolutional layers that consecutively follow kernel sizes 4x4, 4x4 and 7x7 with output channels 64, 32, and 1, respectively.

The discriminator accepts generated mel-spectrograms from the corresponding target decoder and compares them with real mel-spectrograms of the target domain. Its design is similar to the discriminator proposed by DCGAN [14]. It consists of five convolutional layers, each with a stride of 2x2 so that pooling is performed implicitly and optimally. The first four convolutional layers have a kernel size of 4x4 and the last has a kernel size of 3x3. All layers except the last are followed by a LeakyRelu of 0.2, and the second to fourth layers are each followed by instance normalisation.

3.4 Overall Objective and Loss Functions

The overall loss function L that we aim to minimise is additively composed of four individual loss functions, i.e.

$$L = L_{\text{GAN}} + L_{\text{VAE}} + L_{\text{CC}} + L_{\text{Latent}}\,, \tag{1}$$

where the individual loss functions L_{GAN}, L_{VAE}, L_{CC} and L_{Latent} will be defined throughout the remainder of this section. For each training step, the weights of the generator sub-paths (encoder-decoder) are first updated and then the weights of the discriminators are updated. The generators are tuned with respect to reconstructive losses for content embedding (L_{VAE}, L_{CC} and L_{Latent}), and an adversarial loss for trying to generate realistic mel-spectrograms in the target domain. Meanwhile, discriminators depend on an adversarial loss but from the perspective that opposes the generators, where weights are tuned to distinguish generated mel-spectrograms from real mel-spectrograms in the target domain. From these objectives, an overall objective is implied as described by Equation 1.

The adversarial loss L_{GAN} is defined as

$$L_{\text{GAN}} = \lambda_0 \mathbb{E}_{x\sim X}\|D(x)\|^2 + \lambda_0 \mathbb{E}_{z\sim Z}\|1 - D(G(z))\|^2\,. \tag{2}$$

In this notation, $D(x)$ refers to the classification output of the discriminator and λ_0 is a constant, positive hyperparameter for directing the priority of this loss component. More description on constant hyperparameters of each loss component can be found at the end of this section. For each discriminator, an error is minimised such that a mel-spectrogram x (whether fake or real) is correctly identified as belonging to their prior distribution X. Meanwhile, for each generator, an error is minimised such that a mel-spectrogram $G(z)$ generated via a latent code z (with a prior of the latent distribution Z) is identified as real. The discriminators are assessed on classifying mel-spectrograms of the target timbre to enforce realistic target motivation.

As opposed to the initial definition proposed by Goodfellow et al [5], L_{GAN} does not use Binary Cross-Entropy (BCE) for the error criterion but instead uses Mean Squared Error (MSE) as motivated by LSGAN [15]. Between the predicted classifications and ground truth labels, BCE computes the logarithmic difference of probabilities, whereas MSE computes the total difference of squared

magnitudes. MSE is typically used in recent style transfer models as this seems to empirically stabilise the adversarial aspect of the training process. From adopting MSE over BCE, it is worth noting that no sigmoid activation is required as a final layer in the discriminator.

The variational encoding loss L_{VAE}, i.e.

$$L_{\text{VAE}} = \lambda_1 \mathbb{KL}(Z, p(z)) - \lambda_2 \mathbb{E}_{z \sim Z}(\log p_D(x|z)) \tag{3}$$

consists of two terms. The first term is Kullback Leibler (KL) divergence between the latent distribution Z and a zero-mean Gaussian distribution $p(z)$. The latent distribution Z is defined by the output μ of the encoder. The second term is the reconstructive component that aims to successfully recover a mel-spectrogram x from a corresponding latent code z through a probabilistic decoder $p_D(x|z)$. For this, L1 is used as an error criterion as it encourages sparsity which is suitable for mel-spectrograms. L1 is computed between the input source mel-spectrogram and the reconstructed source mel-spectrogram (recovered by feeding z to a decoder from an inverse path).

The cyclic consistency loss L_{CC}, i.e.

$$L_{\text{CC}} = \lambda_3 \mathbb{KL}(Z_{cc}, p(z_{cc})) - \lambda_4 \mathbb{E}_{z_{cc} \sim Z_{cc}}(\log p_D(x|z_{cc}))\,, \tag{4}$$

is computed with the same loss components as L_{VAE} but the estimated latent distribution Z_{CC} and latent code z_{cc} are taken from a cyclic reconstruction. Here, a generated mel-spectrogram is encoded again to obtain Z_{CC} and z_{cc}, and by using the decoder from an inverse path, the source mel-spectrogram is reconstructed. The right term ensures cyclic consistency. Meanwhile, the left term makes it so that the latent space distribution gets modelled with respect to generated mel-spectrograms. Since the encoder is shared and the same weights are tuned for both Z and Z_{CC}, this may serve as an obstacle in modelling a latent space distribution at the bottleneck specific to real mel-spectrograms. It may be worth investigating a model variation where the KL divergence is omitted in L_{CC}.

$$L_{\text{Latent}} = \lambda_5 \|\mu_A - \mu_B\|_1 \tag{5}$$

The latent loss L_{Latent} is the ℓ^1 error between a pair latent means μ_A and μ_B from different source mel-spectrograms A and B respectively. This makes it so that a focus on embedding content is further encouraged at the bottleneck. In the case that a many-to-many variation of this model is pursued, L_{Latent} would have to be calculated between each possible pairing of latent means.

The constant hyperparameters for the loss functions were set such that $\lambda_0 = 10, \lambda_1 = 0.1, \lambda_2 = 100, \lambda_3 = 0.1, \lambda_4 = 100, \lambda_5 = 10$. Here, the reconstructive loss components are favoured over the rest to provide more stability in loss minimisation and to prioritise encoding content before focusing on decoding target mel-spectrograms. Moreover, Adam optimisers [16] were used with a set learning rate of $\alpha = 0.0001$ and running average coefficients $\beta_0 = 0.5$ and $\beta_1 = 0.999$. Learning rate schedules were used to start decaying the learning rate from halfway through the maximum epochs for a given dataset. For training, the batch size was set to 4 samples per timbre domain.

3.5 Inference

After the model is trained, an inference procedure is set up such that it can be applied to input audio of an arbitrary length (provided a length longer than 1.6 seconds). The audio is first preprocessed and a mel-spectrogram is computed with the same method used for training. Since the model only accepts 128x128 mel-spectrograms, a sliding window of a 128 frame length is set to traverse the audio with an overlap count of 4. With each slide, a mel-spectrogram is inferred with the timbre of the target audio. An average of the magnitude values is then taken between the overlapping regions of the transferred mel-spectrograms, resulting in the timbre transfer of a full-length sample.

After the inference, the Fast Griffin Lim algorithm [17] is used to convert the inferred mel-spectrogram to an audio format. Since Griffin Lim is susceptible to phase artefacts, the reconstructed audio will be taken as input to the mel-spectrogram-to-audio vocoder model (in this case, WaveNet) for quality improvement. Here, the reconstructed audio would be preprocessed back to mel-spectrograms with respect to the foreign vocoder's preprocessing steps, then reconstructed once more back to audio but with much fewer phase artefacts. This makes it so that it is not a requirement for the preprocessing specifications of the style transfer model to match that of the vocoder.

3.6 WaveNet Vocoding

An open-source implementation for WaveNet was utilised [18] and separately trained for each of the datasets. One was trained for converting timbres between speakers, and another for converting between instruments. Each vocoder was trained on all timbres per dataset since intonations that may only be present in the source domain and not the target domain should still be considered for style transfer. With dilated convolutions and causal filters, the vocoders successfully learn to generate audio from preprocessed mel-spectrograms and minimise the loss in the audio reconstruction. This results in models of different weights for audio generation general to the timbres from each task domain, which are then used for inferring high-quality audio reconstructions of the target audio generations from the style transfer model.

4 Experiments and Evaluation

4.1 Datasets

The model was trained separately on two datasets; the Flickr 8k Audio dataset [19], and the URMP dataset [20]. From the Flickr dataset, audio files of male and female speakers with the most recordings were used for investigating model variations in the same context as AlBadawy & Lyu [12]. Here, speakers utter a variety of short sentences. Meanwhile, from the URMP dataset, audio files of instruments with the most to least recordings were used for extending the model's application to the context of musical timbre. To mirror the application of voice conversion, only solo recordings of instruments were considered where timbre is monophonic and not polyphonic, though experiments with polyphonic timbre (like in Mor et al [11]) may be worth investigating in the future.

Table 1: Dataset information with respect to each timbre of interest

Dataset	Source	Samples	Total Time	Avg. Time (per sample)
Flickr	Female 1	1686	1 hrs, 48 mins, 1 s	3.6 s
	Male 1	2965	2 hrs, 46 mins, 36 s	3.6 s
	Female 2	1058	58 mins, 9 s	3.6 s
	Male 2	2461	2 hrs, 30 mins, 43 s	3.6 s
URMP	Trumpet	22	35 mins, 36 s	1 min, 32.4 s
	Violin	34	51 mins, 1 s	1 min, 20.4 s
	Flute	18	28 mins, 30 s	1 min, 50 s
	Cello	11	16 mins, 50 s	1 min, 57 s

The number of samples per timbre and time length information were summarised in Table 1. Speakers from Flickr have much more samples than instruments from URMP. On the other hand, the average time per sample is much shorter for Flickr than URMP. Still, the total time recorded of URMP instruments still amounts to significantly less than Flickr speakers. As a result, the URMP experiments should be trained for more epochs such that the total number of steps better match the amount computed for Flickr. Furthermore in Flickr, male 1 has the largest amount of samples whereas female 2 has the smallest. Meanwhile in URMP, violin has the most samples, whereas cello has the least.

4.2 Metrics

Two metrics were utilised for evaluating the two main aspects of the model; SSIM (Structural Similarity Index [21]) for the reconstructive aspect, and FAD (Frechét Audio Distance [22]) for the adversarial translation aspect. Here, SSIM focuses on judging reconstruction in terms of mel-spectrograms of the style transfer model and FAD focuses on comparing generated target audio (after WaveNet) with real target audio.

SSIM is a similarity metric that compares the perceptual quality between an original image and its reconstructed counterpart. This is typically used for assessing image compression algorithms. This is applicable here since mel-spectrograms are two-dimensional and VAEs depend on reconstructing samples post-compression. SSIM is more suitable over other metrics such as PSNR (Peak Signal

Noise Ratio) or MSE as it better considers structure [23] which is especially important in time-frequency representations. By using SSIM to compare a reconstructed mel-spectrogram with its original (after one encoding pass) and separately a cyclic reconstruction with its original (after two encoding passes), the two reconstructive processes of the model are assessed.

FAD takes the approach of FID (Frechét Inception Distance [24]) and adapts it from the context of images to audio. It uses a VGGish model [25] (a variant of the discriminative model VGG [26]) that is pre-trained on a large-scale audio event dataset called AudioSet [27]. Due to the dataset it is trained on, its weights are able to produce semantically rich embeddings of audio. FAD estimates the multivariate Gaussians of VGGish embeddings of real audio samples and separately generated audio samples, then computes the Frechét Distance between them. This effectively estimates the difference between the two data distributions where the smaller the distance, the more realistic the generated set of samples are. As a computed metric, FAD is found more favourable over person dependent metrics such as MOS (Mean Opinion Score) since it is more objective and better replicable.

4.3 Model Experiments

Four versions of the style transfer model were trained for evaluation. The initial version follows the proposed specifications of AlBadawy and Lyu [12]. Meanwhile, the no KLD cyclic version makes it so that KL divergence is not included in the cyclic loss component, making the model focus more on real input for modelling the distribution of the shared latent space. The bottleneck residual version replaces all basic residual blocks with bottleneck residual blocks to investigate the effectiveness of an alternate design with fewer parameters. And finally, the many-to-many version introduces more pathways in training for cyclically transferring between more timbre domains (in this case four) which should further enforce and generalise a shared latent space.

For the Flickr dataset, most style transfer models were trained for 100 epochs and for the URMP dataset, they were trained for 500 epochs (due to the difference in dataset size). Epochs for training the many-to-many experiments were reduced to 17 for Flickr and 84 for URMP since 6 times the amount of steps were pursued per epoch by each network. For each dataset, a WaveNet vocoder was trained for 450,000 steps since plots seemed to feasibly align with input audio plots at this point (as demonstrated in the Appendix Section C). Most of the experiments were one-to-one where the source domain was taken as the timbre opposing the target. Due to time limitations for training, one-to-one experiments were only pursued for the first two selected timbres of each dataset (between female 1 and male 1 for speakers, and trumpet and violin for instruments). For the many-to-many experiments, metrics were calculated in pairs between the first two selected timbres and separately between the last two selected timbres. Unlike in training, timbres were not exhaustively pursued for the evaluation due to the expensive time requirements posed by the inference procedure of WaveNet. Either with more time or another vocoder with a less time costly inference procedure, more timbre pairings may be investigated for further evaluation.

Table 2: SSIM of Reconstructions

Dataset	**Target**	**Model Experiments**			
		Initial	**No KLD Cyclic**	**Bottleneck Residual**	**Many to Many**
Flickr	Female 1	**0.87**	**0.87**	0.79	0.86
	Male 1	0.89	0.88	0.76	**0.91**
	Female 2	0.86	0.85	0.75	**0.89**
	Male 2	0.82	0.83	0.71	**0.87**
URMP	Trumpet	0.93	0.93	0.85	**0.94**
	Violin	0.91	0.91	0.82	**0.92**
	Flute	0.87	0.87	0.73	**0.90**
	Cello	0.86	0.86	0.75	**0.91**

As shown across a variety of timbres, the reconstruction quality of the first reconstruction (Table 2) is consistently higher than the cyclic reconstruction (Table 3). This indicates a subtle loss of information with each transfer since the cyclic reconstruction goes through one more encoding pass and decoding pass than the first reconstruction. As shown in both tables, most of the investigated VAE-GAN variations do not supersede the SSIM of the initial version, but a majority of the many-to-many

Table 3: SSIM of Cyclic Reconstructions

Dataset	Target	Model Experiments			
		Initial	No KLD Cyclic	Bottleneck Residual	Many to Many
Flickr	Female 1	0.73	0.74	0.73	**0.77**
	Male 1	0.80	0.78	0.68	**0.82**
	Female 2	0.76	0.76	0.66	**0.78**
	Male 2	0.70	0.70	0.57	**0.77**
URMP	Trumpet	0.83	0.83	0.78	**0.89**
	Violin	0.81	0.81	0.78	**0.88**
	Flute	0.64	0.65	0.62	**0.82**
	Cello	0.86	0.86	0.66	**0.91**

experiments do. This validates the hypothesis that content-encoding benefits from having a larger variety of source domains and indicates that with more timbres considered, less information is lost in encoding content.

The bottleneck residual experiments demonstrate a considerable lack in reconstruction quality, and so basic residual blocks are more suitable for enriching content information to and from the latent space. The no KLD cyclic experiments achieve reconstruction performance on par with the initial experiments, showing that its inclusion or exclusion does not hold a significant impact on the results. This may be due to the adversarial aspect sufficiently motivating translated data to resemble real data such that no obstructions are made when modelling the bottleneck data distribution for the real data.

Inspecting the many-to-many experiments with respect to each dataset, male 1 achieved the best SSIM for both types of reconstructions. This could likely be attributed to male 1 having the largest data size out of all other timbres (Table 1). Meanwhile for URMP, trumpet achieved the best SSIM for both types of reconstructions. Across the other one-to-one experiments, it is also found that trumpet still achieves the highest SSIM (if not the equivalent). The fact that trumpet has a higher SSIM than violin is surprising since violin has a larger data size than the other instrument timbres. This may indicate that trumpet has a less complex timbre than violin.

Table 4: Frechét Audio Distance (General Vocoding)

Dataset	Target	Model Experiments			
		Initial	No KLD Cyclic	Bottleneck Residual	Many to Many
Flickr	Female 1	2.96	**2.77**	9.10	4.31
	Male 1	1.65	2.48	6.97	**1.40**
	Female 2	**2.35**	**2.35**	8.04	2.64
	Male 2	**1.82**	1.95	7.03	2.90
URMP	Trumpet	**5.26**	5.52	6.06	5.85
	Violin	**4.50**	5.52	12.68	4.99
	Flute	**4.37**	4.40	6.39	5.64
	Cello	20.53	18.20	16.70	**16.21**

FADs computed between the generated audio and real audio of each timbre are presented in Table 4. For each computation, the target timbre's training set was taken as real audio, whereas the transferred audio (post-processed with WaveNet) was taken as the generated audio to test. Contrasting the SSIM results, it shows that the initial experiment performed best for a majority of the transfers. However, for male 1 and cello, the many-to-many experiment performed best. For cello, this suggests that in the case of limited target data, it assists the transfer by using other targets as an assisting resource. For male 1, female 1 was used as the source audio. It is likely that the many-to-many training helped work through the noise present in the recordings of female 1.

Generally, FAD values are worse for instruments than they are for speakers. Also after training the WaveNet vocoders, intonations from mel-spectrograms were reconstructed to less of an accurate degree for instruments. This could be largely attributed to the lack of instrument data relative to the speaker data (Table 1). Griffin Lim reconstructions of cello audio were surprisingly found to sound

more audibly realistic than WaveNet reconstructions. Reflecting the lack of quality, the FAD for cello is significantly worse than the FADs of other timbres. The fact that cello was the timbre with the least data and that no other timbres sounded as bad post-WaveNet implies that WaveNet is sensitive to data size and would greatly benefit from more data for training. Audio signals of instruments can be more complex than speaker signals, and so mel-spectrograms may not capture information as well as possible for instruments. Even though results are audibly more realistic than TimbreTron [10], it could be worth investigating a VAE-GAN design for CQT spectrograms at least for subtle FAD improvements. Another possible future direction would be to investigate other mel-spectrogram vocoders since WaveNet seems better suited for vocoding the audio of speakers than instruments.

4.4 Target Visualisations

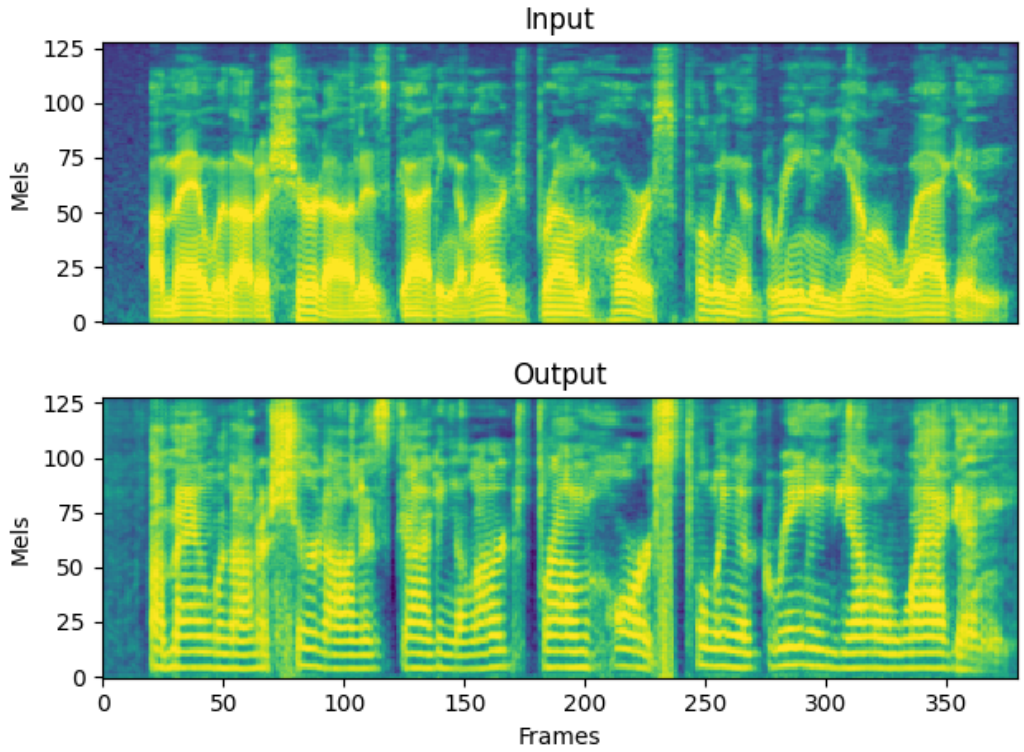


Figure 2: A target visualisation from the many-to-many experiment of female 1, with male 1 as the source input.

Input and output mel-spectrograms of targets female 1 and trumpet were plotted to demonstrate the timbre transfer capabilities of the VAE-GAN model as seen in Figures 2 and 3 (visualisations of other targets may be found in the Appendix Section B). The model retains the content as demonstrated by the vertical spectral features yet modifies the horizontal formants such that they are better suited to the target timbre. From male 1 to female 1, formants are modified such that they are more spaced out and exist less sparsely in a higher range of frequency. From violin to trumpet, formants are introduced at low frequencies, more sparsity is introduced at higher frequencies, and the spacing of formats is modified in a particular way (as can be seen between frames 3,000 and 5,000). These modifications are specific to the target timbres such that their real spectral structure is replicated (of which descriptions and visualisations can be found in Appendix Section A).

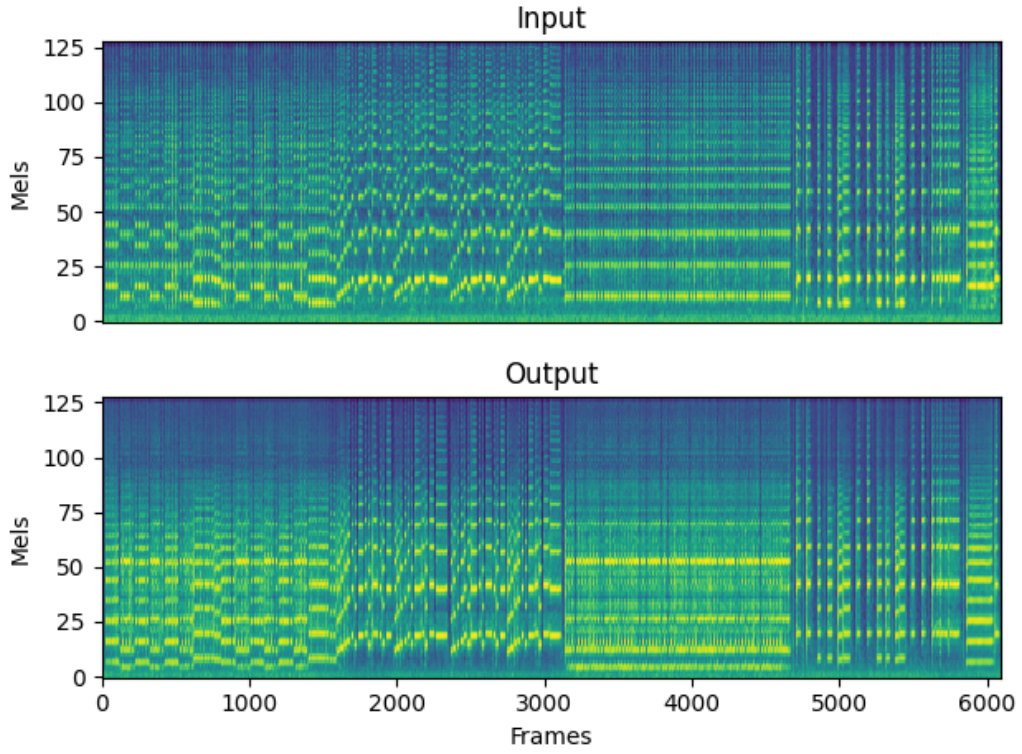


Figure 3: A target visualisation from the many-to-many experiment of trumpet, with violin as the source input.

5 Conclusion and Future Work

In conclusion, a VAE-GAN approach to timbre transfer was not only shown viable in the context of speakers (for voice conversion [12]) but also musical instruments. The instrument timbre transfer results achieved a sufficient audible quality with a relatively simple model working in the time-frequency domain. This model may also be applicable to the transfer of polyphonic timbre in the future since it does not depend on a monophonic pitch transcriptions such as works like [28]. The lack of a dependence on a monophonic pitch transcription likely hurts the quality for instrument timbre transfer (as the audible quality of [28] is evidently higher), but at least this allows the approach to be general enough for application to more than just one type of audio style transfer problem. With more data as well as further design considerations, the audio quality of this approach may be improved.

From the VAE-GAN model experiments, a number of indications were deducted across speakers and instruments; that basic residual blocks supersede bottleneck residual blocks around the bottleneck for enriching content information, that the presence of KL divergence for the cyclic loss component does not significantly impact performance, and finally, that the many-to-many extension outperforms the initial one-to-one version in terms of reconstructive capabilities due to the increased variation of data passed through the universal encoder. Though many-to-many improves the reconstructive aspect of the model, improvements on the adversarial translation aspect were inconclusive. More clarity may be produced by training WaveNet with a more balanced dataset and with more data, or by adopting a different time-frequency vocoder with less sensitivity to data quantity.

References

[1] American National Standards Institute and Acoustical Society of America. *USA Standard, Acoustical Terminology (including Mechanical Shock and Vibration)*. ANSI standard. ANSI, 1976.

[2] Stephen McAdams, Suzanne Winsberg, Sophie Donnadieu, Geert De Soete, and Jochen Krimphoff. Perceptual scaling of synthesized musical timbres: Common dimensions, specificities, and latent subject classes. *Psychological research*, 58(3):177–192, 1995.

[3] EL Saldanha and John F Corso. Timbre cues and the identification of musical instruments. *The Journal of the Acoustical Society of America*, 36(11):2021–2026, 1964.

[4] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *Stat*, 1050:1, 2014.

[5] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Stat*, 1050:10, 2014.

[6] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. *Advances in Neural Information Processing Systems*, 30, 2017.

[7] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[8] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson. Autovc: Zero-shot voice style transfer with only autoencoder loss. In *International Conference on Machine Learning*, pages 5210–5219. PMLR, 2019.

[9] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv*, 2016.

[10] Sicong Huang, Qiyang Li, Cem Anil, Xuchan Bao, Sageev Oore, and Roger B. Grosse. Timbretron: A wavenet(cycleGAN(CQT(audio))) pipeline for musical timbre transfer. In *International Conference on Learning Representations*, 2019.

[11] Noam Mor, Lior Wolf, Adam Polyak, and Yaniv Taigman. A universal music translation network. In *International Conference on Learning Representations*, 2018.

[12] Ehab A. AlBadawy and Siwei Lyu. Voice Conversion Using Speech-to-Speech Neuro-Style Transfer. In *Proc. Interspeech 2020*, pages 4726–4730, 2020.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[14] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv*, 11 2015.

[15] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2017.

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv*, 2014.

[17] Nathanaël Perraudin, Peter Balazs, and Peter L Søndergaard. A fast griffin-lim algorithm. In *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 1–4. IEEE, 2013.

[18] r9y9. Wavenet vocoder implementation, https://github.com/r9y9/wavenet_vocoder, 2018.

[19] David Harwath and James Glass. Deep multimodal semantic embeddings for speech and images. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 237–244, 2015.

[20] Bochen Li, Xinzhao Liu, Karthik Dinesh, Zhiyao Duan, and Gaurav Sharma. Creating a multitrack classical music performance dataset for multimodal music analysis: Challenges, insights, and applications. *IEEE Transactions on Multimedia*, 21(2):522–535, 2019.

[21] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[22] Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. Frechet audio distance: A metric for evaluating music enhancement algorithms. *arXiv*, 2018.

[23] Zhou Wang and Alan C. Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE Signal Processing Magazine*, 26(1):98–117, 2009.

[24] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[25] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. Cnn architectures for large-scale audio classification. In *2017 ieee international conference on acoustics, speech and signal processing (icassp)*, pages 131–135. IEEE, 2017.

[26] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[27] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.

[28] Michael Michelashvili and Lior Wolf. Hierarchical timbre-painting and articulation generation. *arXiv preprint arXiv:2008.13095*, 2020.

A Appendix

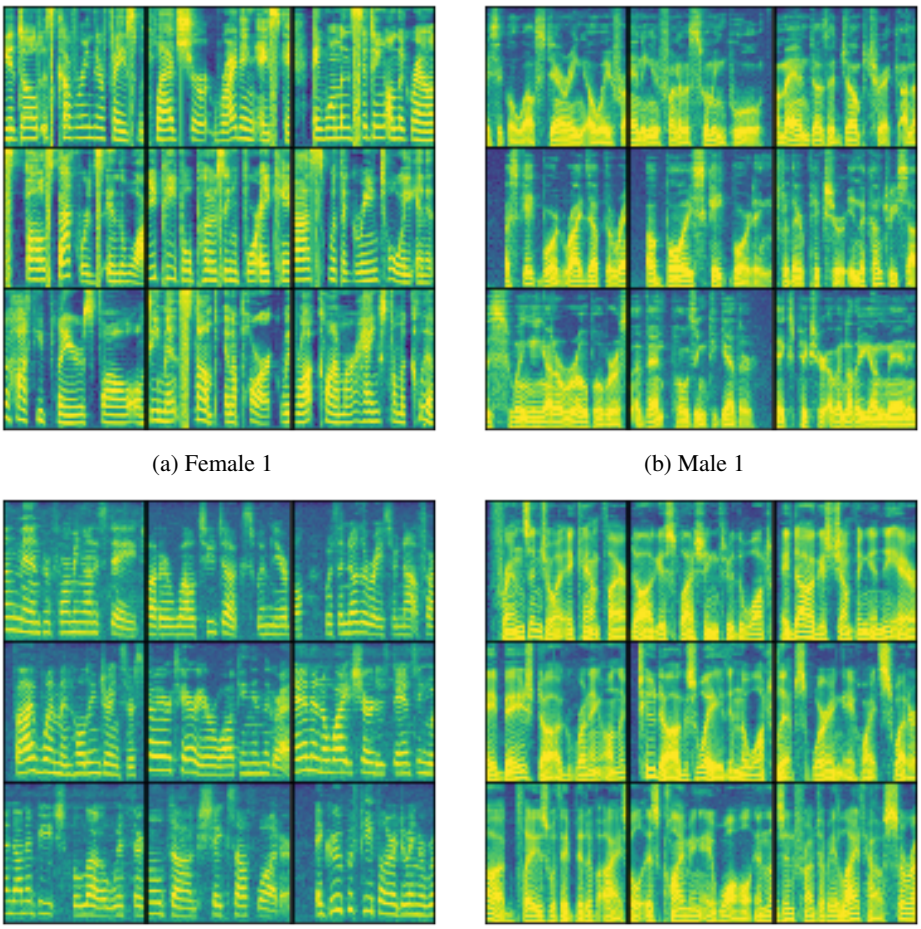
A.1 Dataset Visualisations

A variety of 128x128 mel-spectrogram excerpts were plotted from the selected speakers and instruments to illustrate differences in timbre from the time-frequency perspective of the model (as shown in Figures 4 and 5 respectively). As presented in Figure 4, the horizontal features of females are more vertically spaced out and less compressed to lower regions than that of males. This is reflective of how female voices are higher than males, where the frequency bands exist more prominently over a wider and higher frequency range. There are not many significantly noticeable differences within sex, but female 2 seems to have less intensity in their utterances than female 1.

In comparison to utterances (Figure 4), the horizontal features of the solo musical performances (Figure 5) are much longer due to the slower pace of the audio content. The horizontal features of violin are more prominent in the upper region than that of the other instruments, indicating a higher voicing. The trumpet has an evident sparseness in the upper area, and so does the flute but to slightly less of a degree. The cello has a dense organisation of horizontal features compressed to the lower region. Similar to how features of males were from Flickr, this is indicative of a lower voicing. At a closer inspection, it is also noticeable that the violin and cello sometimes have subtly oscillating frequency bands. This is indicative of vibrato, a technique commonly used in string instruments.

A.2 Extra Target Visualisations

Examples of inferred targets not shown in the main text are presented here in Figures 6-11. By comparing the generated target output to the corresponding mel-spectrograms of real data in Figures 4-5, it can be seen that the VAE-GAN modifies the nature of the spectral bands such that they better match the specified target.



(a) Female 1

(b) Male 1

(c) Female 2

(d) Male 2

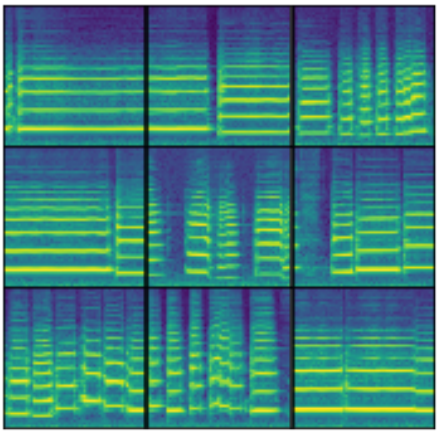
Figure 4: A variety of real mel-spectrogram excerpts per speaker from the Flickr dataset.

A.3 WaveNet Vocoding Visualisations

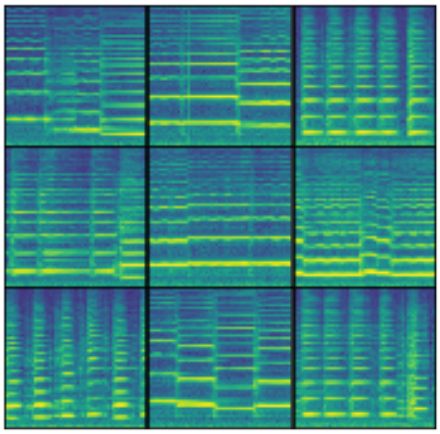
As presented in Figures 12-13, after 450,000 steps two WaveNet vocoders are able to sufficiently reconstruct audio signals from mel-spectrograms for speakers and instruments, respectively. It is worth highlighting that the WaveNet vocoder for instruments reconstructs to less of an accurate degree for attacks and decays than the WaveNet vocoder for speakers as noticable at 0.05s and 0.4s of Figure 13. If this is due to the expressive complexity of the instruments, more steps or data per instrument would be appropriate for training in future experiments. For training with more steps, a faster time-frequency vocoder could also be considered.

A.4 Hardware Specifications

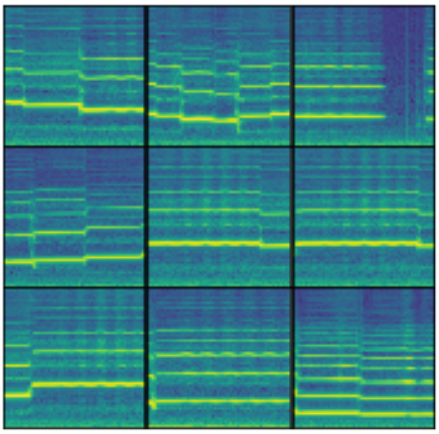
The three most expensive parts of the project from least to most expensive were; the VAE-GAN, the WaveNet vocoder, and the FAD evaluation. In order to execute this project it is recommended to work with NVIDIA GPUs with VRAM as enlisted in Table 5.



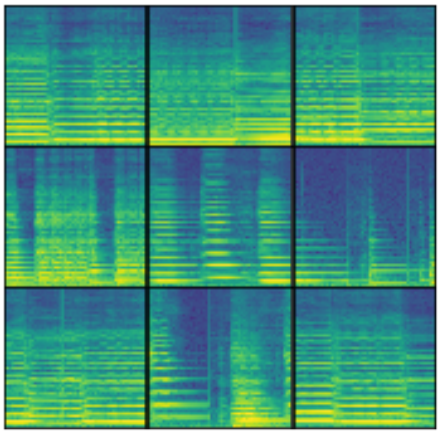
(a) Trumpet



(b) Violin



(c) Flute



(d) Cello

Figure 5: A variety of real mel-spectrogram excerpts per instrument from the URMP dataset.

Table 5: GPU Specifications

Stage	**Recommended VRAM (GB)**
VAE-GAN	2
WaveNet	8
FAD	24

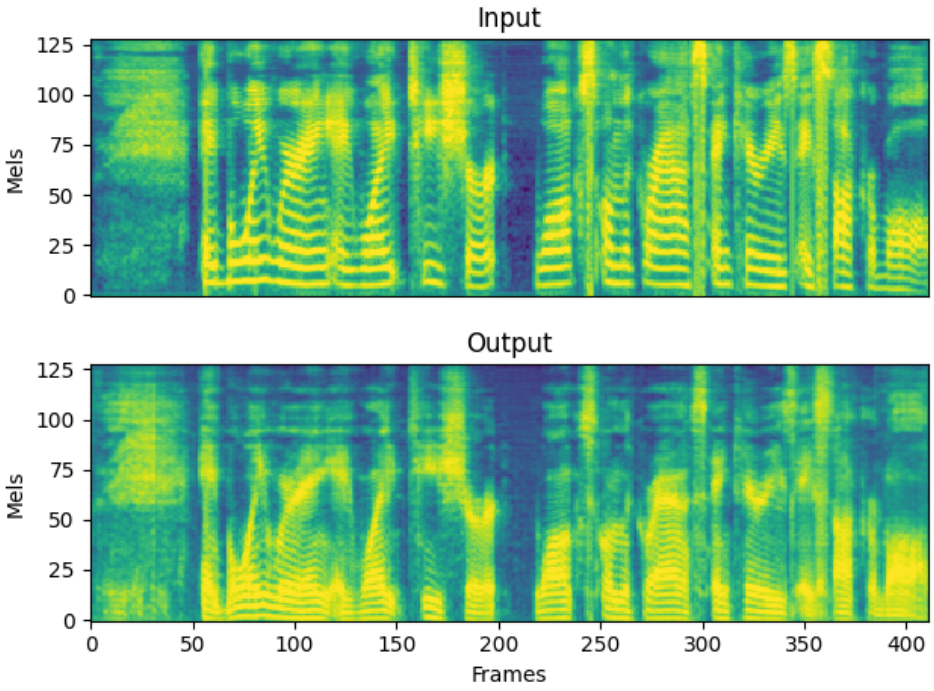


Figure 6: A target visualisation from the many-to-many experiment of male 1, with female 1 as the source input.

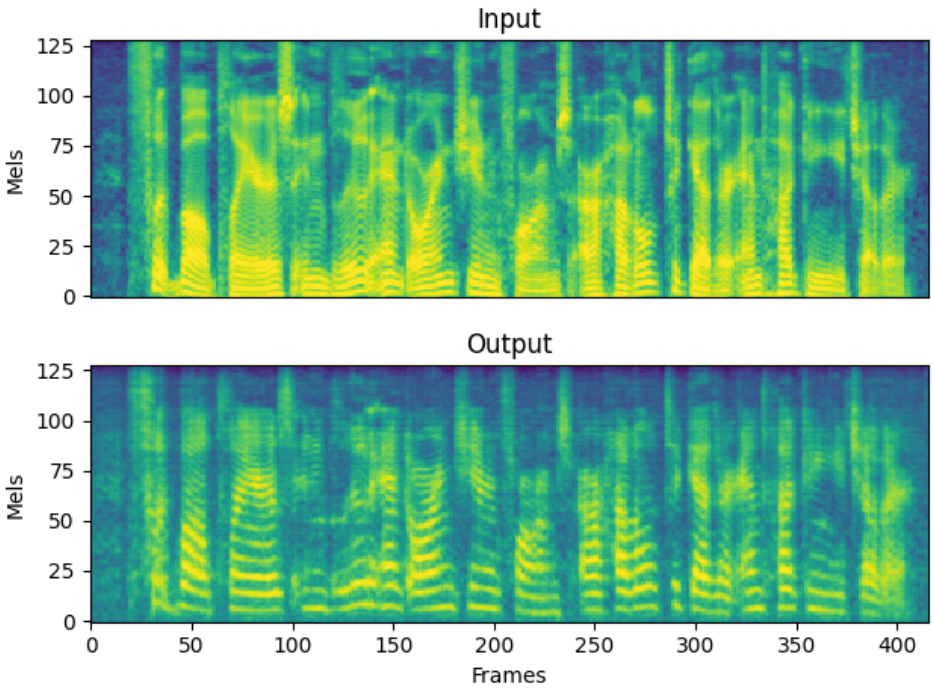


Figure 7: A target visualisation from the many-to-many experiment of female 2, with male 2 as the source input.

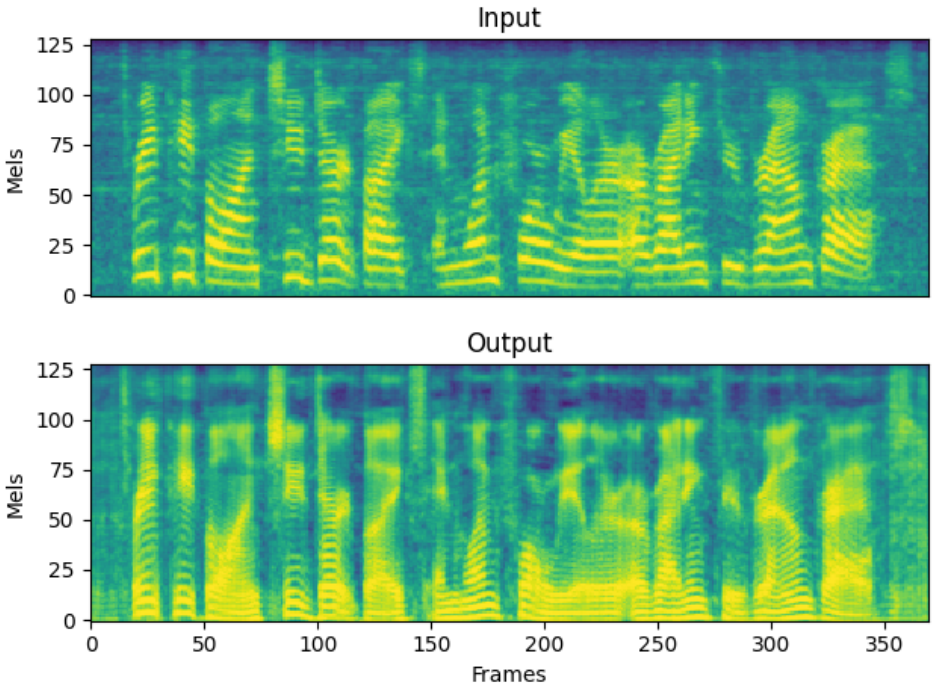


Figure 8: A target visualisation from the many-to-many experiment of male 2, with female 2 as the source input.

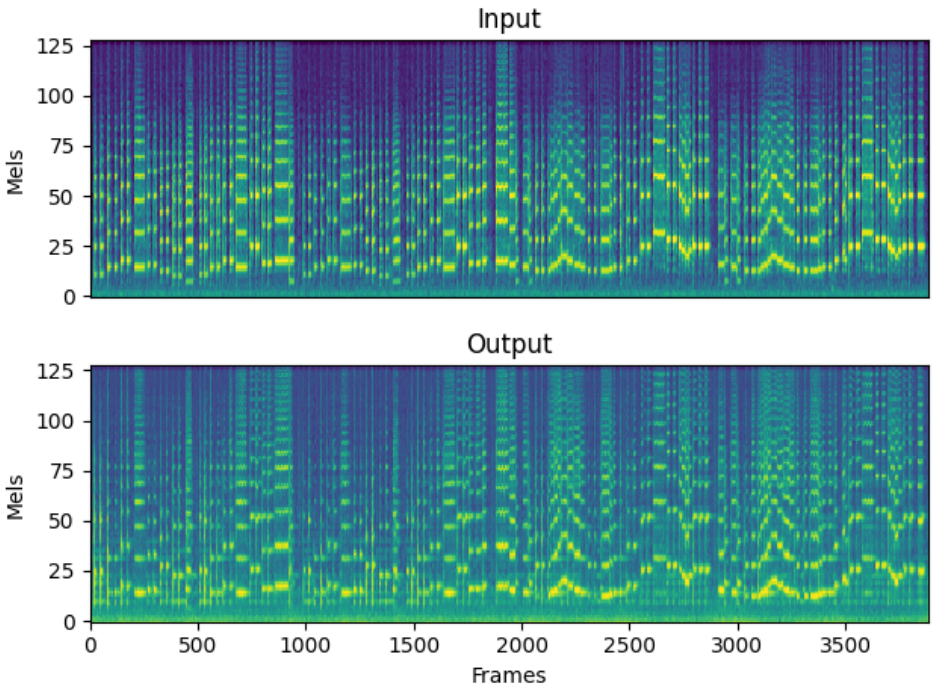


Figure 9: A target visualisation from the many-to-many experiment of violin, with trumpet as the source input.

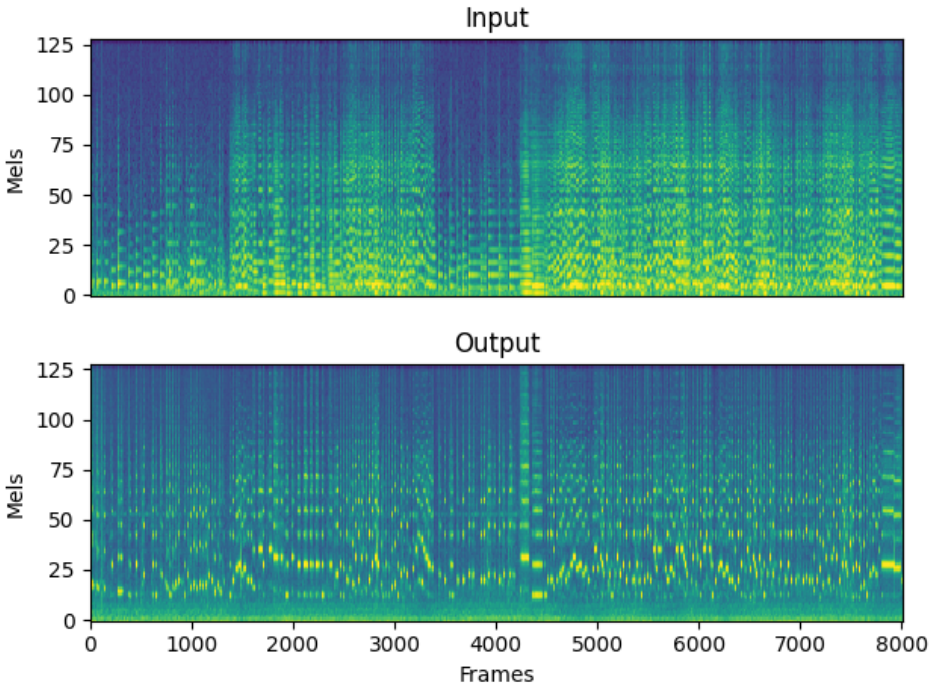


Figure 10: A target visualisation from the many-to-many experiment of flute, with cello as the source input.

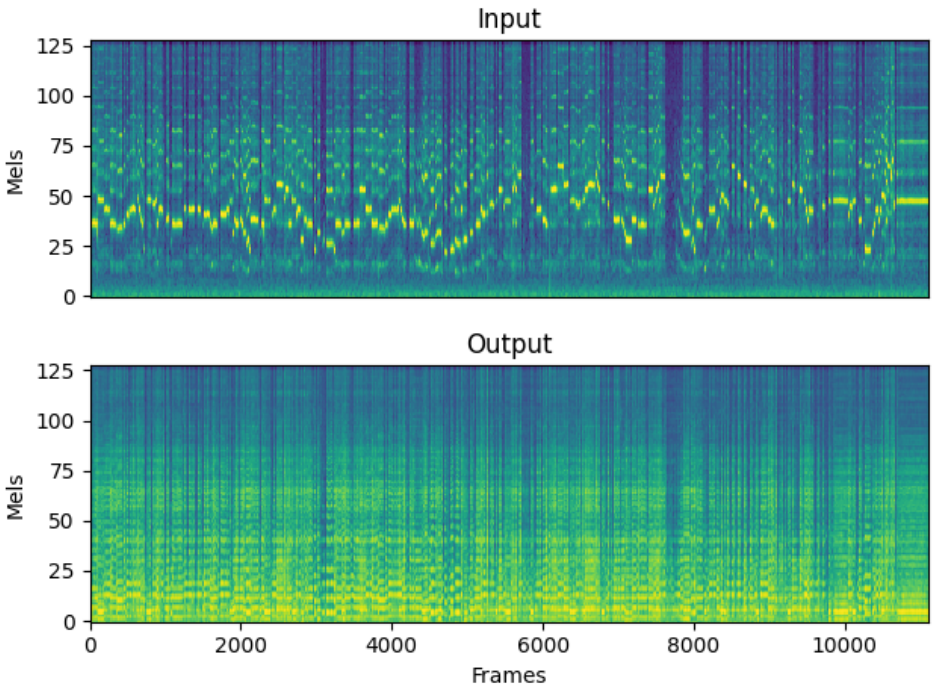


Figure 11: A target visualisation from the many-to-many experiment of cello, with flute as the source input.

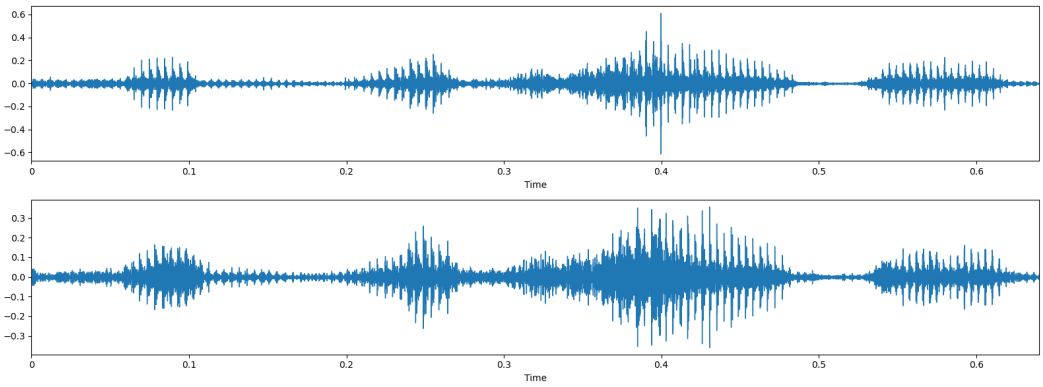


Figure 12: A vocoded reconstruction (bottom) of a real utterance (top) after training on speakers for 450,000 steps.

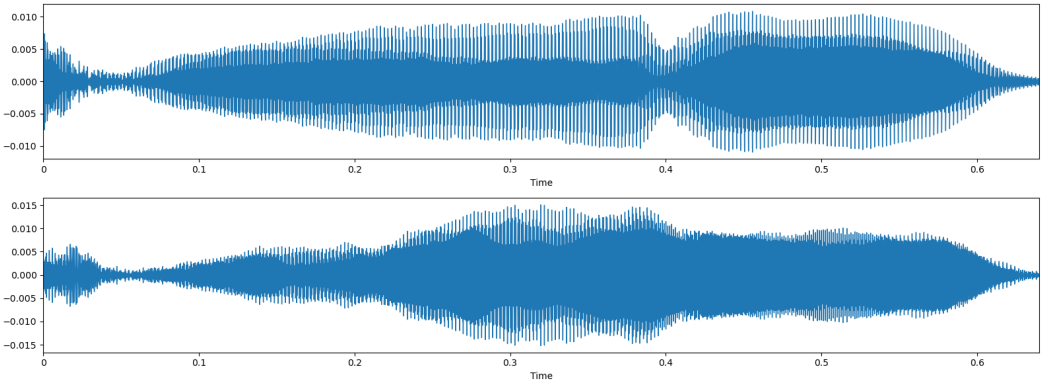


Figure 13: A vocoded reconstruction (bottom) of a real musical recording (top) after training on instruments for 450,000 steps.