LEARNING DISENTANGLED REPRESENTATIONS OF TIMBRE AND PITCH FOR MUSICAL INSTRUMENT SOUNDS USING GAUSSIAN MIXTURE VARIATIONAL AUTOENCODERS

Yin-Jyun Luo[1,2] **Kat Agres**[2,3] **Dorien Herremans**[1,2]
[1] Singapore University of Technology and Design
[2] Institute of High Performance Computing, A*STAR, Singapore
[3] Yong Siew Toh Conservatory of Music, National University of Singapore
yinjyun_luo@mymail.sutd.edu.sg,kat_agres@ihpc.astar.edu.sg,dorien_herremans@sutd.edu.sg

ABSTRACT

In this paper, we learn disentangled representations of timbre and pitch for musical instrument sounds. We adapt a framework based on variational autoencoders with Gaussian mixture latent distributions. Specifically, we use two separate encoders to learn distinct latent spaces for timbre and pitch, which form Gaussian mixture components representing instrument identity and pitch, respectively. For reconstruction, latent variables of timbre and pitch are sampled from corresponding mixture components, and are concatenated as the input to a decoder. We show the model's efficacy using latent space visualization, and a quantitative analysis indicates the discriminability of these spaces, even with a limited number of instrument labels for training. The model allows for controllable synthesis of selected instrument sounds by sampling from the latent spaces. To evaluate this, we trained instrument and pitch classifiers using original labeled data. These classifiers achieve high F-scores when tested on our synthesized sounds, which verifies the model's performance of controllable realistic timbre/pitch synthesis. Our model also enables timbre transfer between multiple instruments, with a single encoder-decoder architecture, which is evaluated by measuring the shift in the posterior of instrument classification. Our in-depth evaluation confirms the model's ability to successfully disentangle timbre and pitch. [1]

1. INTRODUCTION

A disentangled feature representation is defined as having disjoint subsets of feature dimensions that are only sensitive to changes in corresponding factors of variation from observed data [2, 27, 32]. Deep generative models [13, 19, 25, 33] have been exploited to learn disentangled representations in different domains. In the visual domain, studies are focused on learning independent representations for data generative factors such as identity and azimuth [5, 14, 26]. In natural language generation, efforts have been made to generate texts with controlled sentiment [10, 18, 36]. Also in the speech domain, we have witnessed successful attempts in controllable speech synthesis by disentangling factors such as speaker identity, speed of speech, emotion, and noise level [15, 17, 35]. There has been relatively little research on learning disentangled representations for music. In this paper, we disentangle the pitch and timbre of musical instrument sound recordings.

Pitch and timbre are essential properties of musical sounds. Given that one pitch can be played with different instruments, we assume they can be separated. From the perspective of music analysis, disentangled representations of pitch and timbre can be regarded as timbre- and pitch-invariant features which could be exploited for downstream tasks [29,30]. From the synthesis point of view, disentangled representations enable the generation of musical notes with identical pitches (timbres) and different timbres (pitches). Recently, Hung *et al.* presented the first attempt to learn disentangled representations of pitch and timbre for synthesized music by using frame-level instrument and pitch labels based on encoder-decoder networks [21]. Even though the authors managed to change instrumentation to some extent without affecting pitch structure, the approach was restrictive, as it worked with MIDI-synthesized audio and relied on clean frame-level labels, which are scarce to find. Disentangled representations allow for several applications, including music style transfer. Brunner *et al.* proposed a model based on variational autoencoders (VAEs) [25] to generate music with controllable attributes [4]. While genre was factorized by an auxiliary classifier, other musical properties were entangled. Besides the aforementioned models based on MIDI, research on audio has focused on translating between different domains of instrumentation [3,7,20,28]. None of them, however, has addressed learning disentangled latent variables of both pitch and timbre.

This research distinguishes itself from others by disentangling instrument sounds into distinct sets of latent variables (i.e., pitch and timbre), with a framework based on Gaussian Mixture VAEs (GMVAEs). We model the gener-

[1] Example audio files and code at `http://bit.ly/2Dbyt9j`

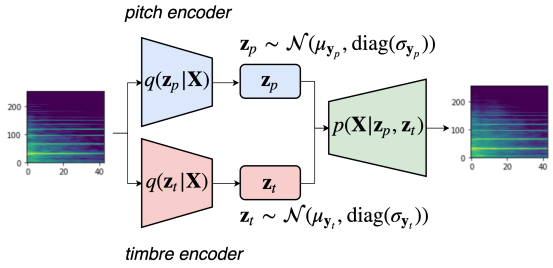

Figure 1. The proposed framework includes separate encoders for pitch and timbre, and a shared decoder.

ative process of an isolated musical note by independently sampling pitch and timbre (instrument) categorical variables. Note that the two factors are actually dependent in a sense that range of pitch is instrument-dependent, however, we verify the model's capability to disentangle them under this simplified assumption of independence. Conditioned on these categorical variables, Gaussian-distributed latent variables are then sampled that characterize variation in the sampled pitch and instrument, respectively. Finally, the data are generated conditioned on the two latent variables. We favor the proposed framework over vanilla VAEs [8, 9] for its more flexible latent distribution compared to a standard Gaussian. In addition, it allows for unsupervised or semi-supervised clustering, which can learn interpretable mixture components and corresponding Gaussian parameters. More importantly, such a framework facilitates the applications in this research: controllable synthesis of instrument sounds, and many-to-many transfer of instrument timbres. Our proposed framework differs from previous studies on timbre transfer, in that we achieve transfer between *multiple instruments* without training a domain-specific decoder for each instrument (e.g. [28]), and we infer both the pitch and timbre latent variable without requiring categorical conditions of source pitch and instrument as in [3]. We evaluate our model by visualizing both the latent space and the synthesized spectrograms, and explore the classification F-scores of classifiers trained in an end-to-end fashion. The results confirm the model's ability to learn disentangled pitch and timbre representations. The rest of the paper is organized as follows: in Section 2, we discuss the proposed framework, and Section 3 describes the dataset and experimental setup. Experiments and results are reported in Section 4. We conclude our work and provide future directions in Section 5.

2. PROPOSED FRAMEWORK

In this section, we briefly describe VAEs and GMVAEs, and elaborate on the proposed framework and architecture.

2.1 Gaussian Mixture Variational Autoencoders

VAEs [25] are unsupervised generative models that combine latent variable models and deep learning [12]. We denote the observed data and the latent variables respectively by $\mathbf{X}$ and $\mathbf{z}$. A graphical model, corresponding to $\mathbf{z} \rightarrow \mathbf{X}$, is trained by maximizing the lower bound of the log marginal likelihood $p(\mathbf{X})$. The intractable posterior $p(\mathbf{z}|\mathbf{X})$ is approximated by introducing a variational distribution $q(\mathbf{z}|\mathbf{X})$ parameterized with neural networks. In regular VAEs, a common choice for the prior distribution $p(\mathbf{z})$ is an isotropic Gaussian, which encourages each dimension of the latent variables to capture an independent factor of variation from the data, and results in a disentangled representation [14]. Such a unimodal prior, however, does not allow for multi-modal representations. GMVAEs [6,22,24] extend the prior to a mixture of Gaussians, and assume the observed data are generated by first determining the mode from which it was generated, which corresponds to learning a graphical model $\mathbf{y} \rightarrow \mathbf{z} \rightarrow \mathbf{X}$. This introduces a categorical variable $\mathbf{y}$, and $q(\mathbf{y}|\mathbf{X})$, which infers the classes of data. This enables semi-supervised learning [24] and unsupervised clustering [6, 22] in deep generative models. In the speech domain, Hsu *et al.* used two mixture distributions to separately model the supervised speaker and unsupervised utterance attributes, which allowed for extra flexibility in conditional speech generation [17]. We build upon this idea to learn separate latent distributions to represent the pitch and timbre of musical instrument sounds. More importantly, to facilitate downstream creative applications such as controllable synthesis and instrument timbre transfer in music, we propose to model supervised pitch representations and semi-supervised timbre representations, with labels of pitch and instrument identity. As such, the mixture components in latent space of pitch and timbre can be clearly interpreted as the classes, i.e., pitch and instrument identity.

2.2 Model Formulation

The latent variables of pitch and timbre for an isolated musical note $\mathbf{X}$ are denoted as $\mathbf{z}_p$ (*pitch code*) and $\mathbf{z}_t$ (*timbre code*), respectively. To represent Gaussian mixture latent distributions, two categorical variables are introduced: an M-way categorical variable $\mathbf{y}_p$ for pitch, where M is the number of recorded pitches in the dataset, and a K-way categorical variable $\mathbf{y}_t$ for timbre, where K is the number of instrument classes. We consider $\mathbf{y}_p$ to be observed (fully supervised), which assumes the availability of pitch labels during training, and is reasonable as we model isolated instrument sounds in this research. For $\mathbf{y}_t$, we investigate both unsupervised and semi-supervised learning, i.e., using varying numbers of instrument labels for training. It is shown in Section 4 that our model can efficiently leverage the limited number of labels. Without loss of generality, we denote $\mathbf{y}_t$ as unobserved (unsupervised) as in [17]. The joint probability of $\mathbf{X}$, $\mathbf{y}_t$, $\mathbf{z}_t$ and $\mathbf{z}_p$ is written as:

$$p(\mathbf{X}, \mathbf{y}_t, \mathbf{z}_t, \mathbf{z}_p|\mathbf{y}_p) = p(\mathbf{X}|\mathbf{z}_p, \mathbf{z}_t)p(\mathbf{z}_p|\mathbf{y}_p)p(\mathbf{z}_t|\mathbf{y}_t)p(\mathbf{y}_t), \quad (1)$$

where $p(\mathbf{y}_t)$ is uniform-distributed, i.e., we do not assume to know the instrument distribution in the dataset. Both the conditional distributions $p(\mathbf{z}_p|\mathbf{y}_p) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{y}_p}, \mathrm{diag}(\boldsymbol{\sigma}_{\mathbf{y}_p}))$ and $p(\mathbf{z}_t|\mathbf{y}_t) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{y}_t}, \mathrm{diag}(\boldsymbol{\sigma}_{\mathbf{y}_t}))$ are assumed to be diagonal-covariance Gaussians with learnable means and constant variances. This amounts to both the marginal prior $p(\mathbf{z}_p)$ and $p(\mathbf{z}_t)$ being Gaussian mixture models

(GMMs) with diagonal covariances. Ideally, each mixture component in the former (*pitch space*) uniquely represents the pitch of $\mathbf{X}$, while that in the latter (*timbre space*) is interpreted as the instrument identity. As we will see in Section 4.1, however, moderate supervision is essential to learn a timbre space that groups instruments perfectly. For creative applications such as the synthesis and timbre transfer of instrument sounds, the proposed model has numerous merits: 1) the learnt representations are not restricted to be unimodal, which offers a more discriminative timbre space than regular VAEs (Section 4.1 and 4.2); 2) direct and intuitive sampling from pitch and timbre space allows for consistent and controllable synthesis of instrument sounds, attributed to the fact that Gaussian parameters of each interpretable mixture component are readily available after training (Section 4.3); and 3) simple arithmetic manipulations between means of mixture components facilitate many-to-many transfer between instrument timbres (Section 4.4). For the training objective, we closely follow the derivation in [17] and train the model by maximizing the evidence lower bound (ELBO) as follows:

$$\begin{aligned}\mathcal{L}(p,q;\mathbf{X},\mathbf{y}_p) = &\mathbb{E}_{q(\mathbf{z}_p|\mathbf{X})q(\mathbf{z}_t|\mathbf{X})}[\log p(\mathbf{X}|\mathbf{z}_p,\mathbf{z}_t)] \\ &- D_{KL}(q(\mathbf{z}_p|\mathbf{X})||p(\mathbf{z}_p|\mathbf{y}_p)) \\ &- \mathbb{E}_{q(\mathbf{y}_t|\mathbf{X})}[D_{KL}(q(\mathbf{z}_t|\mathbf{X})|p(\mathbf{z}_t|\mathbf{y}_t)] \\ &- D_{KL}(q(\mathbf{y}_t|\mathbf{X})||p(\mathbf{y}_t)),\end{aligned} \tag{2}$$

where $p(\mathbf{X}|\mathbf{z}_p,\mathbf{z}_t)$, $q(\mathbf{z}_p|\mathbf{X})$, and $q(\mathbf{z}_t|\mathbf{X})$ are parameterized with neural networks, referred to as the *decoder*, *pitch encoder* [2] , and *timbre encoder*, respectively. Instead of using another neural network, we approximate $q(\mathbf{y}_t|\mathbf{X})$ by $\mathbb{E}_{q(\mathbf{z}_t|\mathbf{X})}[p(\mathbf{y}_t|\mathbf{z}_t)]$. Readers interested in detailed derivation are referred to Appendix A in [17].

2.3 Architecture

Our model is composed of a shared decoder and separate encoders for pitch and timbre, as illustrated in Figure 1. Specifically, we reshape the T-by-F spectrogram to have number of channels $C = F$, each of which is a T-by-1 vector, where T and F refer to time and frequency. Each encoder contains two one-dimensional convolutional layers, each with 512 filters of shape 3×1, and a fully connected layer with 512 units. A Gaussian parametric layer follows and outputs two $L-$dimensional vectors which represent mean and log variance. $\mathbf{z}_p$ and $\mathbf{z}_t$ are sampled from the Gaussian layer with the reparameterization trick [25], which enables stochastic gradient descent, and are then concatenated for the decoder to reconstruct the input. The architecture of the decoder is symmetric to the encoder. Batch normalization followed by the activation function `relu` are used for every layer except for the Gaussian and the output layer. We use the activation function `tanh` for the output layer as we normalize the data within $[-1, 1]$.

[2] A common alternative is conditioning the model with categorical pitch labels such that one does not have to train a pitch encoder [3,7]. It, however, requires the pitch of the inputs to be known a priori to performing tasks such as timbre transfer [3], and also prohibits the model from extracting pitch features for downstream tasks. By training this extra encoder, we also demonstrate how one can extend the model to possibly learn multiple interpretable latent variables.

3. EXPERIMENTAL SETUP

In this section, we describe the experimental setup, including details of the dataset, input representations, and model configurations.

3.1 Dataset

Inspired by Esling *et al.* [8], we use a subset of Studio-On-Line (SOL) [1], a database of instrument note recordings. [3] The dataset contains 12 instruments, i.e, piano (`Pno`, 246), violin (`Vn`, 138), cello (`Vc`, 147), English horn (`Ehn`, 128), French horn (`Fhn`, 214), tenor trombone (`Trtb`, 63), trumpet (`Trop`, 194), saxophone (`Sax`, 99), bassoon (`Bn`, 251), clarinet (`Clr`, 180), flute (`Fl`, 118) and oboe (`Ob`, 107). There are 1,885 samples in total. All recordings are resampled to 22,050Hz, and only the first 500ms segment ($T = 43$) of each recording is considered. We extract Mel-spectrograms with 256 filterbanks ($F = 256$), derived from the power magnitude spectrum of the short-time Fourier transform (STFT). To compute STFT, we use a Hann window with window size of 92ms and hop size of 11ms. As a result, the input representation is a 43-by-256 Mel-spectrogram. The dataset is split into a training (90%) and validation set (10%), each containing the same distribution of instruments. The magnitude of the Mel-spectrogram is scaled logarithmically, and the minimum and maximum values in the training set are used for normalizing the magnitude within $[-1, 1]$ in a corpus-wide fashion to preserve differences in dynamics.

3.2 Hyperparameters

In order to train both the GMMs in pitch and timbre space, we initialize the means of mixture components using Xavier initialization [11]. We set constant standard deviations, rather than trainable ones, for pitch and timbre space. For pitch space, $\boldsymbol{\sigma}_{\mathbf{y}_p} = \mathbf{e}^{-\mathbf{2}}$ for all mixture components, which is relatively small, as each mixture component represents a pitch, and we do not expect a large variance over recordings that play the same pitch. For timbre space, we let $\boldsymbol{\sigma}_{\mathbf{y}_t} = \mathbf{e}^{\mathbf{0}}$ for all mixture components, which captures the timbre variation of each mixture component, i.e., instrument identity. The dimensionality of the latent space is $L = 16$, and the numbers of mixture components are $M = 82$ and $K = 12$, equivalent to the numbers of classes of pitch and instrument, respectively. For all experiments, a batch size of 128 is used, model parameters are initialized with Xavier initialization and are trained using the Adam optimizer [23] with a learning rate of 10^{-4}.

In addition to the proposed model (M_{GMVAE}), we consider a baseline (M_{VAE}) that substitutes the timbre space with an isotropic Gaussian as in regular VAEs. Training such a model amounts to optimizing Eqn (2) with the last two terms replaced with $D_{KL}(q(\mathbf{z}_t|\mathbf{X})||p(\mathbf{z}_t))$, where $p(\mathbf{z}_t) = \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. The experimental results in Section 4.1 and Section 4.2 show that $\mathrm{M_{GMVAE}}$ learns a more discriminative and disentangled timbre space than $\mathrm{M_{VAE}}$.

[3] Access to the dataset was requested from [8].

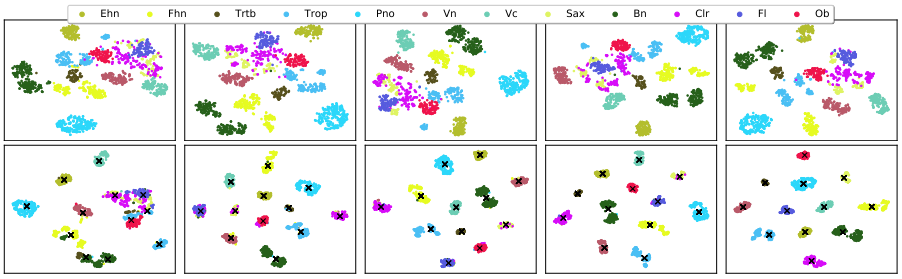

Figure 2. Timbre space visualization of $\mathrm{M}_{\mathrm{VAEs}}$ (top) and $\mathrm{M}_{\mathrm{GMVAEs}}$ (bottom). From left to right: models trained with 0, 25, 50, 75, or 100% of instrument labels, respectively.

	Instrument Classification					Pitch Classification				
N (%)	CNN	$\mathrm{M}_{\mathrm{VAE}}$		$\mathrm{M}_{\mathrm{GMVAE}}$		CNN	$\mathrm{M}_{\mathrm{VAE}}$		$\mathrm{M}_{\mathrm{GMVAE}}$	
		$\mathbf{z}_t$	$\mathbf{z}_p$	$\mathbf{z}_t$	$\mathbf{z}_p$		$\mathbf{z}_t$	$\mathbf{z}_p$	$\mathbf{z}_t$	$\mathbf{z}_p$
0	-	0.960	0.163	0.937	0.175	-	0.112	0.966	0.146	0.960
25	0.920	0.960	0.192	0.971	0.180	-	0.169	0.966	0.084	0.977
50	0.983	0.971	0.169	0.988	0.186	-	0.158	0.977	0.079	0.977
75	1.000	0.971	0.169	1.000	0.163	-	0.079	0.971	0.045	0.977
100	1.000	0.937	0.158	1.000	0.197	0.983	0.039	0.983	0.028	0.966

Table 1. The F-scores of instrument and pitch prediction by linear classifiers and CNNs. N (%) refers to the percentage of instrument labels used to train the models. Columns $\mathbf{z}_t$ and $\mathbf{z}_p$, respectively, refer to the F-scores obtained using the learned timbre and pitch code to train the down-stream linear classifier.

3.3 Semi-Supervised Learning

We exploit a moderate number of instrument labels to learn a timbre space in which the clusters clearly represent instrument identity. Similar to Kingma *et al.* [24], in the semi-supervised training for $\mathrm{M}_{\mathrm{GMVAE}}$, we *guide* the inference of instrument labels $q(\mathbf{y}_t|\mathbf{X})$ by leveraging limited amounts of supervision. This is done by adding an additional loss term which measures the cross entropy between the inferred and true instrument labels. Because we do not infer $\mathbf{y}_t$ in $\mathrm{M}_{\mathrm{VAE}}$, we use $\mathbf{z}_t$ to train an auxiliary classifier to predict $\mathbf{y}_t$. It has two 128-unit fully-connected layers, and is jointly optimized with $\mathrm{M}_{\mathrm{VAE}}$. We consider varying numbers of instrument labels N = 0 (unsupervised), 25, 50, 75, and 100% (fully supervised) of the total number. We randomly sample and let the label distribution match the distribution of instruments.

4. EXPERIMENTS AND RESULTS

The experiments and the results are presented in this section. We first visualize the timbre space, and quantitatively evaluate the disentangled representations. We then demonstrate the applications of controllable synthesis and many-to-many timbre transfer. Finally, we identify the particular latent dimension that is sensitive to the distribution of the spectral centroid, which allows for finer timbre controls.

4.1 Visualization

Figure 2 visualizes the timbre space using t-distributed stochastic neighbor embedding (t-SNE) [34], a technique that projects vectors from high- to low-dimensional space. We first observe that $\mathrm{M}_{\mathrm{GMVAE}}$ learns a Gaussian-mixture distributed timbre space, with means of mixture components marked as crosses in the figure. Second, attributed to the pitch encoder which addresses pitch variations, both $\mathrm{M}_{\mathrm{VAE}}$ and $\mathrm{M}_{\mathrm{GMVAE}}$ are able to form clusters of instrument identity even without being trained with instrument labels (the leftmost column). We observe that the wind family (e.g., saxophone, clarinet and flute) forms an ambiguous cluster. Such an ambiguity remains in the $\mathrm{M}_{\mathrm{VAE}}$ even with increased N, while it is less present in the $\mathrm{M}_{\mathrm{GMVAE}}$ latent space, due to the multi-modal prior distribution. As we will confirm in Section 4.2, $\mathrm{M}_{\mathrm{GMVAE}}$ outperforms $\mathrm{M}_{\mathrm{VAE}}$ in learning a more discriminative and disentangled timbre space. Note that in $\mathrm{M}_{\mathrm{GMVAE}}$, $p(\mathbf{y}_t)$ is assumed to be uniformly distributed over 12 classes of instruments, i.e., mixture components are equally weighted. As a result, instruments with larger within-class variances (e.g., bassoon and trumpet) are assigned to more than one cluster when $N = 0$. In future work we aim to improve the performance of the unsupervised clustering of instruments.

4.2 Pitch and Instrument Disentanglement

A disentangled pitch (timbre) representation should be discriminative for pitch (instrument identity), and at the same time non-informative of instrument identity (pitch). Therefore, we evaluate $\mathbf{z}_p$ and $\mathbf{z}_t$ by means of classification. We train linear classifiers to map $\mathbf{z}_p$ and $\mathbf{z}_t$ to predict both pitch and instrument labels with one fully connected layer. For comparison, we train an end-to-end convolutional neural network (CNN), whose architecture is the same as the encoder and is a strong baseline, to map the original input Mel-spectrograms to either pitch or instrument labels.

Table 1 shows the results. The CNN achieves high F-scores on both instrument and pitch classification; note that N is the supervisory percentage of the total number of *instrument* labels, and we always use all pitch labels to train the models, which is reasonable as we model isolated notes

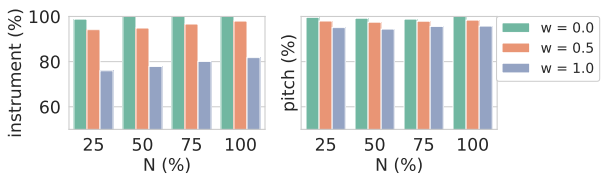

Figure 3. The F-scores for predicting instrument (left) and pitch (right) labels from the synthesized spectrograms.

in this work. In instrument classification, using $\mathbf{z}_t$ as the feature representations outperforms $\mathbf{z}_p$ by a large margin, as expected. Specifically, in both models, the $\mathbf{z}_t$ learned with unsupervised learning ($N = 0$) is already discriminative enough to predict instruments with linear classifiers. While the F-score of M_{GMVAE} improves with increased N, that of M_{VAE} does not. Moreover, the linear classifier trained with $\mathbf{z}_t$ outperforms the CNN when $N < 75$. The timbre space of M_{GMVAE} displays the most discriminative power among the models. We attribute the F-scores of instrument classification attained by $\mathbf{z}_p$ to the fact that the piano covers all possible pitches in the dataset, while other instruments account for a smaller pitch range. As a result, $\mathbf{z}_p$ of notes that were only recorded by piano are correctly classified. Future work can be done to decorrelate particular pitches and instruments by data augmentation and adversarial training as in [16]. In pitch classification, $\mathbf{z}_p$ outperforms $\mathbf{z}_t$ as expected, and both models achieve comparable results. More importantly, M_{GMVAE} performs better than M_{VAE} in terms of disentanglement, as $\mathbf{z}_t$ results in lower F-scores when predicting pitch with increased N.

4.3 Controllable Synthesis of Instrument Sounds

As shown in Figure 2, M_{GMVAE} learns a timbre space $p(\mathbf{z}_t)$, whose mixture components are clearly interpreted as instrument identity when trained with moderate supervision. Meanwhile, mixture components in $p(\mathbf{z}_p)$ represent pitch. As Gaussian parameters are readily available after training, we can achieve controllable sound synthesis by sampling $p(\mathbf{z}|\mathbf{y})$. To synthesize the target pitch y_m and instrument y_k, we first sample $\mathbf{z}_p \sim \mathcal{N}(\boldsymbol{\mu}_{y_m}, w \cdot \text{diag}(\boldsymbol{\sigma}_{y_m}))$ and $\mathbf{z}_t \sim \mathcal{N}(\boldsymbol{\mu}_{y_k}, w \cdot \text{diag}(\boldsymbol{\sigma}_{y_k}))$, where the multiplier $w \in \{0, 0.5, 1.0\}$ serves to examine the effect of sampling latent variables that deviate from the modes. The decoder then synthesizes the Mel-spectrogram by consuming $[\mathbf{z}_t, \mathbf{z}_p]$. For evaluation, the CNNs (trained on the original dataset) are used to test whether the synthesized spectrograms are still recognized as belonging to the desired instrument and pitch. High F-scores therefore indicate high controllability of the model in sound synthesis. We use the sound samples in the validation set as the targets to synthesize, and repeat the sampling 30 times for each target.

The F-scores for pitch and instrument classification are reported in Figure 3. We first note that increasing w degrades classification performance. This is expected, as a sample which is synthesized using a latent variable far from its corresponding mean of mixture component deviates more from the intended instrument or pitch distribution. Moreover, the fact that the CNN was trained on the original samples while tested on the synthesized ones also contributes to the inferior performance. Second, increasing N improves instrument classification performance. Finally, the high F-scores across all N's when $w \in \{0, 0.5\}$ indicate accurate and consistent synthesis of instrument sounds with intended pitches and instruments, even with a timbre space trained using a limited number of instrument labels. This implies that M_{GMVAE} efficiently exploits the instrument labels, and learns a discriminative mixture distribution of timbre, which is consistent with the visualization in Figure 2 (bottom row, $N \geq 25$). We do not explore the timbre space resulting from unsupervised learning ($N = 0$) in this experiment, as the instrument identity of each mixture component is not directly available. We can, however, infer the instrument identity of each mixture component by sampling and synthesis, and expect reasonably good performance for controllable synthesis if the clustering of instruments shown in the bottom left of Figure 2 is improved. This will be explored in future work.

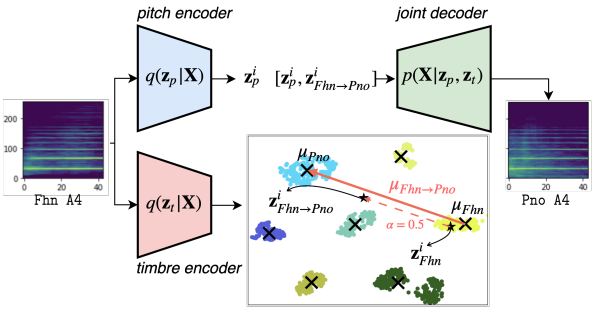

Figure 4. Many-to-many timbre transfer. The ith sample of the `Fhn` is transferred to the `Pno`, with vector arithmetic in the (partially shown) timbre space.

4.4 Many-to-Many Transfer of Timbre

In this experiment, we demonstrate many-to-many transfer of instrument timbre. In Mor *et al.*, a domain-specific decoder was trained for each target [28]. To achieve timbre transfer with a single encoder-decoder architecture, Bitton *et al.* proposed to use a conditional layer [31] which takes both instrument and pitch labels as inputs [3]. On the other hand, our model infers $\mathbf{z}_t$ and $\mathbf{z}_p$, and only uses a single joint decoder. As illustrated in Figure 4, timbre transfer is achieved by decoding $[\mathbf{z}_{transfer}, \mathbf{z}_p]$, i.e., transferring timbre while keeping pitch unchanged, where $\mathbf{z}_{transfer} = \mathbf{z}_{source} + \alpha\boldsymbol{\mu}_{source\to target}$, $\boldsymbol{\mu}_{source\to target} = \boldsymbol{\mu}_{target} - \boldsymbol{\mu}_{source}$, and $\alpha \in [0, 1]$. Once again, we rely on the trained CNNs in Table 1 for evaluation. More specifically, we examine the posterior shift in instrument prediction of the CNN, before and after transferring from source to target instruments with $\alpha = \{0, 0.25, 0.5, 0.75, 1.0\}$. For simplicity, the most frequent instruments (i.e., French horn, piano, cello, and bassoon) of the four families are selected as the representatives, and we perform timbre transfer using the samples in the validation set as the source. For example, consider `Fhn` as the source and `Pno` as target, as shown in Figure 4. We modify the timbre code as $\mathbf{z}_{Fhn\to Pno}^i = \mathbf{z}_{Fhn}^i + \alpha\boldsymbol{\mu}_{Fhn\to Pno}$, where $\mathbf{z}_{Fhn}^i$ is the timbre code of the ith `Fhn` sample, and $i = \{1, 2, \ldots, \text{N}_{\text{Fhn}}\}$.

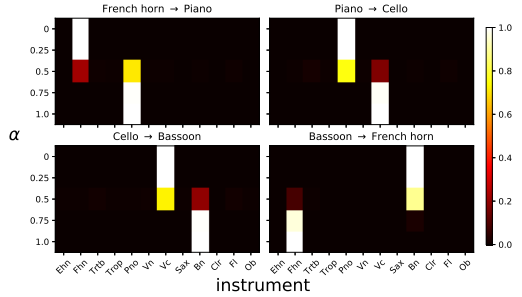

Figure 5. The averaged posterior (color) shift in instrument prediction of the CNN, caused by timbre transfer.

We decode as described earlier and report the averaged posterior (over N_{Fhn}) of instrument prediction of the CNN.

For simplicity, in Figure 5, we report the results of the source-target pairs $Fhn \rightarrow Pno$, $Pno \rightarrow Vc$, $Vc \rightarrow Bn$ and $Bn \rightarrow Fhn$. Each subfigure refers to a source-target pair, and represents the averaged posterior shift of instrument classification of the CNN, with varying α. For all pairs, the biggest posterior shift (hence the prediction change) happens when $\alpha = 0.5$. This also applies to the rest of the possible instrument pairs not shown in the figure. Meanwhile, by using pitch classification, we examine if the pitches are the same before and after timbre transfer, and we use the original pitch labels as ground-truths. We find that, except in the case where the source is piano, all source-target pairs attain a perfect F-score in terms of pitch. This confirms the ability of the model to successfully perform many-to-many timbre transfer. A special case arises when piano is the source. The F-scores before transfer, after transfer to French horn, to cello, and to bassoon, are 0.958, 0.750, 0.791, and 0.791, respectively. As described earlier in Section 4.2, lower F-scores can be attributed to the fact that the range of piano is much larger than that of the target instruments, or the classifier fails to predict the synthesized samples that have unseen combinations of pitch and instrument. The other possible reason is the model falls short of generalization. Nevertheless, this only happens in some cases when the source is piano; as demonstrated in Figure 6, the model is able to transfer `Pno G6` to cello (the first row), which is an example of generalizing to an out-of-range pitch for the target instrument. In the first and third row, the high-frequency components appear with increased α, and the energy distributes over the segment without decay. The model, however, falls short in

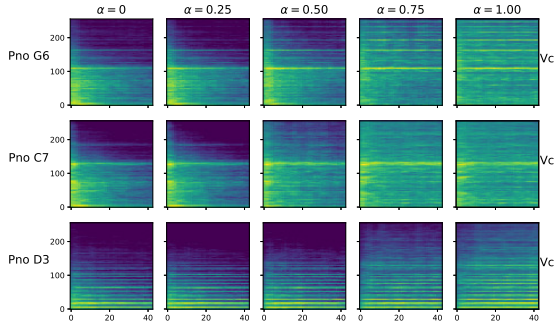

Figure 6. Examples of timbre transfer $Pno \rightarrow Vc$. The top two rows are tones outside of the cello range.

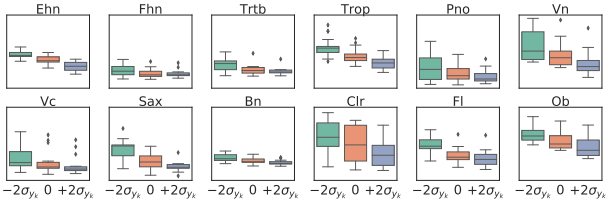

Figure 7. Spectral centroid values in response to z_t^{13}.

generalizing to the higher pitch, i.e., `Pno C7` (the second row), where the energy remains focused at the onset, and high-frequency components are smeared. In the future, we could improve the model generalizability by performing data augmentation and adversarial training as in [16].

4.5 Spectral Centroid Disentanglement

A diagonal-covariance Gaussian prior encourages the model to learn disentangled latent dimensions [14]. This applies to all mixture components in our model. In particular, we identify a latent dimension that correlates with the spectral centroid. we modify the 13th dimension of $\mathbf{z}_t$, z_t^{13}, of each sound sample in the validation set by $\pm 2\sigma_{y_k}$, where $\sigma_{y_k} = e^0$ for all instruments, and then synthesize the spectrograms, for which we then calculate the spectral centroid. Figure 7 shows the distributions of the spectral centroid before and after the modifications. The two-tailed t-test indicates significant differences ($p < 0.05$) between $-2\sigma_{y_k}$ and $+2\sigma_{y_k}$ for all instruments. As demonstrated in Figure 8, we observe that increased z_t^{13} reduces the energy of high-frequency components and results in lower spectral centroid values. In future research, we will further investigate disentangling specific acoustic features for finer control of sound synthesis beyond pitch and instrument.

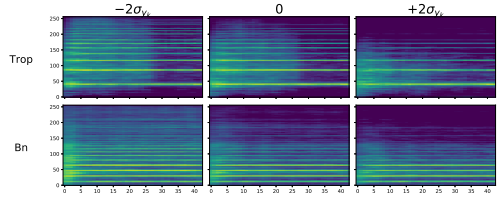

Figure 8. Latent dimension traverse of z_t^{13}.

5. CONCLUSIONS AND FUTURE WORK

We have proposed a framework based on GMVAEs to learn disentangled timbre and pitch representations for musical instrument sounds, which is verified by our experimental setup. We demonstrate its applicability in controllable sound synthesis and many-to-many timbre transfer. In future work, we plan to conduct listening tests for a more comprehensive evaluation of the applications, and further disentangle both low- (e.g., acoustic features) and high-level (e.g., playing techniques) sound attributes, enabling finer control of synthesized timbres. By using supervised and unsupervised learning in a deep generative model, the framework can be easily adapted to learn interpretable mixtures such as singer identity, music style, emotion, etc., which facilitates music representation learning and creative applications.

6. ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their constructive reviews. This work is supported by a Singapore International Graduate Award (SINGA) provided by the Agency for Science, Technology and Research (A*STAR), under reference number SING-2018-01-1270.

7. REFERENCES

[1] G. Ballet, R. Borghesi, P. Hoffmann, and F. Levy. Studio online 3.0: An internet "killer application" for remote acess to ircam sounds and processing tools. *Journee Informatique Musicale*, 1999.

[2] Y. Bengio. Deep learning of representations: Looking forward. In *International Conference on Statistical Language and Speech Processing*, pages 1–37. Springer, 2013.

[3] A. Bitton, P. Esling, and A. Chemla-Romeu-Santos. Modulated variational auto-encoders for many-to-many musical timbre transfer. *arXiv preprint arXiv:1810.00222*, 2018.

[4] G. Brunner, A. Konrad, Y. Wang, and R. Wattenhofer. Midi-vae: Modeling dynamics and instrumentation of music with applications to style transfer. In *Proc. of the International Society for Music Information Retrieval Conference*, pages 23–27, 2018.

[5] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2172–2180, 2016.

[6] N. Dilokthanakul, P. A. M. Mediano, M. Garnelo, M. C.-H. Lee, H. Salimbeni, K. Arulkumaran, and M. Shanahan. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*, 2016.

[7] J. Engel, C. Resnick, A. Roberts, S. Dieleman, M. Norouzi, D. Eck, and K. Simonyan. Neural audio synthesis of musical notes with wavenet autoencoders. In *Proc. of the International Conference on Machine Learning*, pages 1068–1077, 2017.

[8] P. Esling and A. Bitton. Bridging audio analysis, perception and synthesis with perceptually-regularized variational timbre spaces. In *Proc. of the International Society for Music Information Retrieval Conference*, 2018.

[9] P. Esling, A. ChemlaRomeu-Santos, and A. Bitton. Generative timbre spaces with variational audio synthesis. In *Proc. of the International Conference on Digital Audio Effects*, 2018.

[10] Z. Fu, X. Tan, N. Peng, D. Zhao, and R. Yan. Style transfer in text: Exploration and evaluation. In *AAAI*, 2017.

[11] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AAAI*, pages 249–256, 2010.

[12] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 2014.

[14] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, M. Shakir, , and A. Lerchner. Beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

[15] W.-N. Hsu, Y. Zhang, and J. Glass. Unsupervised learning of disentangled and interpretable representations from sequential data. In *Advances in Neural Information Processing Systems*, pages 1878–1889, 2017.

[16] W.-N. Hsu, Y. Zhang, R. J. Weiss, Y.-A. Chung, Y. Wang, Y. Wu, and J. Glass. Disentangling correlated speaker and noise for speech synthesis via data augmentation and adversarial factorization. In *Advances in Neural Information Processing Systems*, 2018.

[17] W.-N. Hsu, Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Y. Wang, Y. Cao, Y. Jia, Z. Chen, J. Shen, P. Nguyen, and R. Pang. Hierarchical generative modeling for controllable speech synthesis. In *International Conference on Learning Representations*, 2019.

[18] Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P Xing. Toward controlled generation of text. In *International Conference on Machine Learning*, pages 1587–1596, 2017.

[19] Z. Hu, Z. Yang, R. Salakhutdinov, and E. P. Xing. On unifying deep generative models. *arXiv preprint arXiv:1706.00550*, 2017.

[20] S. Huang, Q. Li, C. Anil, X. Bao, S. Oore, and R. B. Grosse. Timbretron: A wavenet (cyclegan (cqt (audio))) pipeline for musical timbre transfer. *arXiv preprint arXiv:1811.09620*, 2018.

[21] Y.-N. Hung, Y.-A. Chen, and Y.-H. Yang. Learning disentangled representations for timber and pitch in music audio. *arXiv preprint arXiv:1811.03271*, 2018.

[22] Z. Jiang, Y. Zheng, H. Tan, B. Tang, and H. Zhou. Variational deep embedding: An unsupervised and generative approach to clustering. In *International Joint Conference on Artificial Intelligence*, 2017.

[23] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[24] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, pages 3581–3589, 2014.

[25] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *International Conference on Learning Representations*, 2014.

[26] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. Tenenbaum. Deep convolutional inverse graphics network. In *Advances in Neural Information Processing Systems*, pages 2530–2538, 2015.

[27] A. Kumar, P. Sattigeri, and A. Balakrishnan. Variational inference of disentangled latent concepts from unlabeled observations. *International Conference on Learning Representations*, 2018.

[28] N. Mor, L. Wolf, A. Polyak, and Y. Taigman. A universal music translation network. *arXiv preprint arXiv:1805.07848*, 2018.

[29] M. Muller, D. P. W. Ellis, A. Klapuri, and G. Richard. Signal processing for music analysis. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1088–1110, 2011.

[30] M. Muller and S. Ewert. Towards timbre-invariant audio features for harmony-based music. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3):649–662, 2010.

[31] E. Perez, F. Strub, H. D. Vries, V. Dumoulin, and A. Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

[32] K. Ridgeway. A survey of inductive biases for factorial representation-learning. *arXiv preprint arXiv:1612.05299*, 2016.

[33] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[34] L. van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[35] Y. Wang, D. Stanton, Y. Zhang, RJ Skerry-Ryan, E. Battenberg, J. Shor, Y. Xiao, F. Ren, Y. Jia, and R. A. Saurous. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. *arXiv preprint arXiv:1803.09017*, 2018.

[36] H. Zhou, M. Huang, T. Zhang, X. Zhi, and B. Liu. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *AAAI*, 2017.