Music Style Transfer: A Position Paper

Gus G. Xia
Computer Science Department
New York University Shanghai
gxia@nyu.edu

Shuqi Dai
Computer Science Department
Peking University
shuqid.pku@gmail.com

Abstract

Led by the success of *neural style transfer* on visual arts, there has been a rising trend very recently in the effort of *music style transfer*. However, "music style" is not yet a well-defined concept from a scientific point of view. The difficulty lies in the intrinsic *multi-level* and *multi-modal* character of music representation (which is very different from image representation). As a result, depending on their interpretation of "music style", current studies under the category of "music style transfer", are actually solving completely different problems that belong to a variety of sub-fields of Computer Music. Also, a vanilla end-to-end approach, which aims at dealing with all levels of music representation at once by directly adopting the method of image style transfer, leads to poor results. Thus, we vitally propose a more scientifically-viable definition of music style transfer by breaking it down into precise concepts of timbre style transfer, performance style transfer and composition style transfer, as well as to connect different aspects of music style transfer with existing well-established sub-fields of computer music studies. In addition, we discuss the current limitations of music style modeling and its future directions by drawing spirit from some deep generative models, especially the ones using unsupervised learning and *disentanglement* techniques.

Introduction

Background of Automated Music Generation

The practice of music automation can be traced back to *Guido D'Arezzo*, a famous medieval musician who designed a rule-based vowel-to-pitch mapping algorithm to generate a sequence of notes (Loy 1989). While "crafting music" is still the mainstream, *algorithmic composition*, or in general *automated music generation* has become more and more popular nowadays with the development of modern computers. On the one hand, fast CPUs offer dramatic speedup of experimentations, so that people can test different ideas much more rapidly. In addition, various computer-music programming languages (Dannenberg 1997; McCartney 1996; Boulanger 2000; Wang and Cook 2003) have been invented since the late 1950s, which further boosted the efficiency of music creation via programming. On the other hand, advanced computational models and data-driven algorithms have empowered computers to generate more human-like music via inheriting certain statistics and styles from the training sets. Recently, with the breakthroughs in artificial neural networks, *deep generative models* have become one of the leading techniques for automated music generation (Briot, Hadjeres, and Pachet 2017). For the examples of mimicking J.S. Bach alone, we have seen BachBot (Liang 2016), DeepBach (Hadjeres and Pachet 2016), CNNBach (Huang et al. 2017), etc., and most of them can generate convincing results.

Despite these promising progress, people still struggle to generate both *natural* and *creative* music through automation. In general, algorithms with weak constraints are often "too random" and rarely make human-like music, though many works are interesting and creative from a contemporary perspective. On the other hand, algorithms with strong constraints (either explicitly constrained via rules or implicitly constrained by training data) are mostly "too flat" and lack the exploration and dynamic that can be easily sensed from genuinely creative works.

Music Style Transfer: Importance & Challenges

Image style transfer techniques (Gatys, Ecker, and Bethge 2015) inspired the hope to solve the paradox above. By separating and recombining music contents and music styles of different pieces, it is possible to generate new music that is both creative and human-like. In other words, we can still use our favorite data-driven algorithms but twist the constraints or optimizations in general by applying them separately to different aspects (i.e., content and style) of music.

Such effort is named after *music style transfer*. However, there is a severe problem: "music style" is a fuzzy term that can literally refer to any aspect of music, ranging from high-level compositional features (such as tonality and chord sequence) to low-level acoustic features (such as sound texture and timbre). This ambiguity is mainly due to the intrinsic *multi-level*, *multi-modal* character of music representation — music can be read, listened to, or performed, and it all depends on whether we are relying on *score* (the top-level, abstract representation), *sound* (the bottom-level, concrete representation), or *control* (the intermediate representation). This is very different from image representation, and so far

no end-to-end system can deal with all levels of music representation together in an elegant manner.

Consequently, most studies only focus on a certain level/modality of music representation and therefore have different interpretations of music style. Depending on the interpretation, the essence of music style transfer also varies a lot and may even refer to problems evolved from different sub-fields of computer music, such as algorithmic composition, expressive performance, or sound synthesis. In other words, we are facing an issue of the many-to-one collapse of keyword definition. Without further action, an accumulated upcoming literature all named after "music style transfer" would lead to a great confusion of the underlying problems to the readers as well as a risk to ignore the treasures in computer music before the age of deep learning.

In this position paper, we contribute a precise definition of music style transfer based on the uniqueness of music representation. We start from an overview of music representation in Section 2 in order to formally introduce the definition in Section 3, where we also connect different types of music style transfer with existing well-established computer music studies. In the end, we discuss the current limitations and possible future directions of music style modeling by inspecting current unsupervised learning and disentanglement techniques of deep generative models.

Multi-level and Multi-modal Representation

Music is widely considered a universal language and there are many previous discussions on music representations (Dannenberg 1993; Wiggins, Müllensiefen, and Pearce 2010; Müller et al. 2013). The relationship between music notation (score) and actual sound is similar to the one between text and speech. Score serves as a symbolic and highly-abstract visual representation to efficiently record and communicate music ideas, whereas the sound is a set of continuous and concrete signal representations that encode all the details we can hear. Therefore, we can picture the two representations at different levels, with the score at the top and sound at the bottom (Dannenberg 1993).

In the middle, people often insert an intermediate representation of performance control. The reasons are twofold. First, musical semantics and expression rely heavily on performance control that a funeral hymn can sound really happy by simply tripling the tempo. Second, the performance control for many instruments (e.g., a piano keyboard) can be easily parameterized and therefore very machine friendly. Note that different levels of representation are not solely mutually exclusive, but the multi-level property offers us a useful tool to better understand the essence of music *content* and *style*. To fully comprehend different aspects of music style transfer, we shall first investigate the multi-level property of music representation more in-depth.

Score Representation

Score representation exists in many forms, including sheet music notation, lead sheet, chord chart and numbered musical notation. Most of them are highly symbolic and encode abstract music features indicated by the composer, including

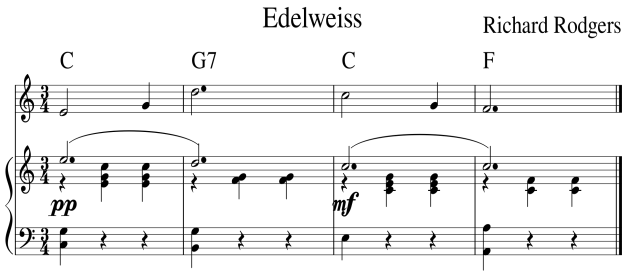

Figure 1: An example of western music notation.

tonality, chord, pitch, timing, dynamics and rich structure information such as phrases and repetitions.

The key character of score representation is that the encoded features are mostly *discrete* with a mix of measurement scale. Take western music notation (Figure 1) for example. *Note onset* is a ratio variable and lies on integer multiples of a certain time unit (usually 1/8 beat is short enough). Pitch is an interval variable, whose corresponding fundamental frequency always lies in a discrete sequence. (E.g., the frequency of C4 in the equal-tempered tuning is 261.63 Hz, the frequency of its successive pitch C♯4 is 277.18 Hz, and there is no other pitch frequencies lie in between.) Dynamics is an ordinal variable, usually ranging from ***ppp***(the softest) to ***fff***(the loudest). Many other symbols are nominal variables, such as chord types and repeat signs. Such characters bring a challenge for generative models since discrete optimization is in general very difficult and a mixed scale makes some numerical operations impossible.

Performance Control Representation

A performance control encodes an interpretation of the corresponding score, rely on which a performer turns the score into performance motions. A commonly used control representation is MIDI piano roll (Figure 2), where each note is encoded by its pitch, dynamics, onset (starting time), and duration. It also has a number of controllers such as pedal and pitch bend for more performance nuances. To be specific, pitches are integers in semitones with C4 being 60, dynamics are integers in velocities units (speed with which the keys are hitting) ranging from 1 to 127, and timings are floating point numbers in seconds.

Compared to score representation, the key character of performance control is the enriched and detailed timing and dynamics information, which more or less determined the *musical expression* of a performance. On the other hand, most structural information such as phrase, repetition, and chord progression is flattened and become implicit during the translation from the score to performance control. Note that performance control is largely independent from the actual instrument; it is not yet the final music sound and still considered a middle-level abstraction.

Sound Representation

Sound, the concrete signal representation, can be seen as an acoustic realization of the corresponding performance con-

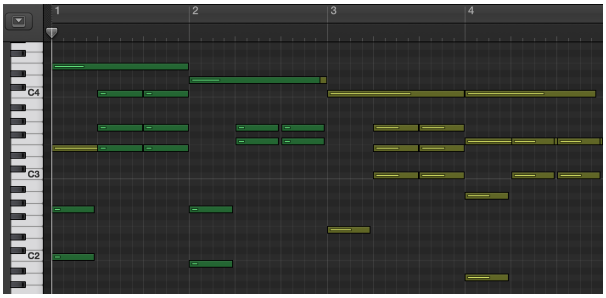

Figure 2: An example of MIDI piano roll representation.

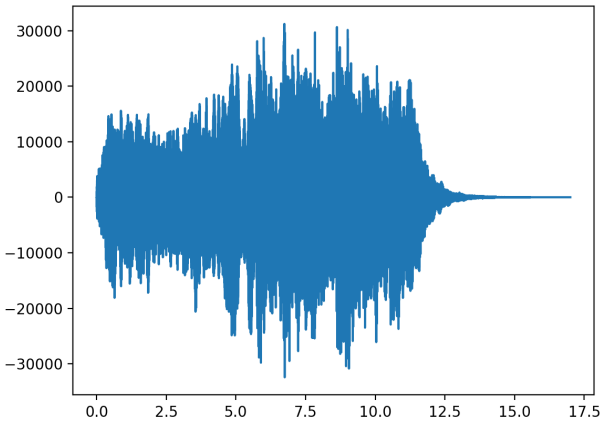

Figure 3: A waveform example where the horizontal axis represents time and the vertical axis represents amplitude.

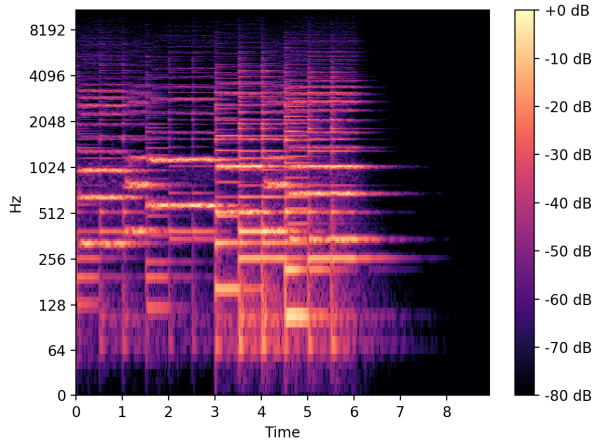

Figure 4: A spectrogram example where the horizontal axis represents time, vertical axis represents frequency, and the color represents energy distribution on different frequencies.

trol via a certain instrument. Two commonly used formats for sound representation are waveform (Figure 3) and spectrogram (Figure 4).

The key character of sound representation is purely *continuous* and rich in acoustic details such as timbre, articulation, and other nuances not available in other levels of representation. At the expense of such acoustic details, all symbolic abstractions together with precise performance control information become no more explicit and get hidden in the audio.

Representation, Content, and Style

Table 1 shows a summary of different music representations. It is important to notice that the multi-level architecture actually has already implied the essence of music content and music style, i.e., *music content is the information extracted through abstraction (from a lower level to a higher level), while music style is the information enriched through interpretation and realization (from a higher level to a lower level).*

Table 1: A summary of music representations.

	Sensory system	Unique features	Scale of measure	Type of data
Score (top)	visual	structure & symbolic abstractions	all	discrete
Control (middle)	motor	expressive timing & dynamics	interval & ratio	mixed
Sound (bottom)	auditory	acoustic details	ratio	continuous

Thus, a complete end-to-end system for music style transfer should at least fulfill three requirements: 1) be cross-modal and flexible to deal with different measurement scales, 2) automatically extract the performance control and score information from a sound input, and 3) freely manipulate music representations at any level. However, we have to accept the fact that such systems do not yet exist and may not emerge in the near future. The second requirement alone remains an open problem (especially for polyphonic music), and has been the main focus of the whole field of *music information retrieval* for many years.

Therefore, it is beneficial to first solve style transfer for each level of music representation and gradually integrate different components into one system. A hasty attempt at an end-to-end music style transfer system by directly adopting the algorithms for image style transfer (Dmitry and Vadim 2016; Gao 2017) would only lead to results that sound like a casual remix of different pieces of music.

Music Style Transfer: A Precise Definition and Related Work

We present the precise definitions of music style transfer for each level of representation in a bottom-up order. They are: 1) *timbre style transfer* for sound, 2) *performance style transfer* for performance control, and 3) *composition style transfer* for score. We also include a brief overview of the related work and connect them with existing sub-fields of computer music.

Timbre Style Transfer

Definition 1: *Timbre style transfer applies to sound representation. It means to alter the timbre information in a meaningful way while preserving the hidden content of performance control.*

A successful timbre style transfer would allow us to reproduce a trumpet performance by a flute with the same musical expression. Timbre style transfer is closely related to *sound synthesis* (Russ 2004), especially the studies aiming to synthesize different sound of acoustic instruments. The difference is that timbre style transfer requires a *disentanglement* of timbre (style) and performance control (content) and implies that there is room to create new timbre through the combination of different ones.

Two recent pioneer studies on timbre style transfer are Google's WaveNet autoencoders (Engel et al. 2017) and Stanford's audio spectrograms neural style transfer system (Verma and Smith 2018). The former built an autoencoder for raw waveform using WaveNet (a dilated temporal convolutional neural network). The bottleneck hidden layer is therefore considered a timbre representation and used to create new timbre through linear interpolation. The latter treated audio spectrograms as images and applied image style transfer with additional carefully designed constraints on temporal and frequency energy envelopes.

We shall also see the limitations. For both works, the disentanglement of timbre and performance control information is not yet very successful, especially when the length of the processed audio unit is long. Also, from a synthesis perspective, the sound quality of synthesized instruments is still far from the state-of-art learning-based synthesis techniques (Hu 2004) and worth further investigation. As a side note, VisualSoundtrack (Ananthabhotla and Paradiso 2017), which is named after "style transfer", is actually a synthesis system. It requires human inputs of pitch and no disentanglement is involved.

Performance Style Transfer

Definition 2: *Performance style transfer applies to performance control representation. It means to alter the control information in a meaningful way while preserving the implicit score content.*

A successful performance style transfer would allow us to transfer Louis Armstrong's interpretation of Summertime to the one of Miles Davis. It is closely related to *expressive performance rendering*, which studies how to convert static scores into human-like expressive performances by different computational models. (Kirke and Miranda 2009; Widmer and Goebl 2004; Simon and Oore 2017) The difference is that performance style transfer requires a disentanglement of control (style) and score information (content) and implies that there is room to create new musical expression through the combination of different controls.

As far as we know, there is no work on performance style transfer yet, though performer identification (Ramirez, Maestre, and Serra 2010; Stamatatos and Widmer 2005) has been studied for over a decade. One close attempt is the recent Duet Interaction system (Xia 2016) that can generate an expressive accompaniment based on the performance style of a solo, but it requires a pre-defined score and cannot create new performance styles. As a side note, the work named after "neural translation of musical style" (Malik and Ek 2017) is actually an expressive performance rendering system, which focuses on dynamic generation given a score input. Thus, performance style transfer remains a brand-new field worth exploring.

Composition Style Transfer

For many forms of score, there is room for further abstraction. Take western music notation for example, the most identifiable score feature, in general, is the melody contour and sometimes with the structural functions of harmony (Schoenberg and Stein 1969). This is especially the case for tonal music.

Definition 3: *Composition style transfer means to preserve the identifiable melody contour (and the underlying structural functions of harmony) while altering some other score features in a meaningful way.*

A successful composition style transfer would allow us to create *variation*, *improvisation*, *re-harmonization*, or *rearrangement* of a piece of music. A representative masterpiece is the *Twelve Variations on "Ah vous dirai-je, Maman"* by Mozart. Take the first variation for example, it mostly preserved the melody contour and chord progression of the theme and altered the rhythm and texture to a large extent. Recent high-quality pieces (made by human) include: *Improvisation of "Mary had a little lamb"*[1], a Korean style *Carmen Overture*[2], and a Chinese style Mozart Sonata[3]. Composition style transfer is closely related to *stylistic automatic composition*, which can be traced back to David Cope (Cope and Mayer 1996). The difference between these two topics is that composition style transfer requires a disentanglement of different score features and implies that there is room to create new types/idioms of score features (such rhythm, texture, and chord progression) through the combination of different ones.

Pioneer studies on automatic composition style transfer include (Pati 2018; Zalkow 2016; Kaliakatsos-Papakostas et al. 2017), where the first two deal with monophonic composition and the last one deals with polyphonic composition. The work (Pati 2018) builds pitch and rhythm models separately for different music genres and then create new

[1] https://youtu.be/Q6Usd3_fbq8

[2] https://youtu.be/hKv2_UCo1ZQ

[3] https://soundcloud.com/wang-michael-452158298/sets/style-transform-of-mozart-sonata

melodies through the combination of the pitch model of one genre and the rhythm model of another genre. The works by (Zalkow 2016; Kaliakatsos-Papakostas et al. 2017) rely on the power of explicit rules to modify melody and merge different chord progressions, respectively. The work (Lattner, Grachten, and Widmer 2016) enforces certain music structures by considering additional template-matching constraints in the optimization procedure.

The demo pieces created by these early studies are still quite immature, especially compared to the pieces made by humans. The major problem is actually not "how to transfer the composition style" but "how to model it" in the first place. Current composition models still lack the capacity or representation of music structure and the hidden "grammar" of chord progressions. Note that most successful cases of the automatic stylistic composition are related to Bach, and at least for non-experts the structure of Bach's compositions is rather local and easy to perceive compared to many other composers. This is unlikely to be a coincidence and worth the attention of future studies.

Future Directions of Music Style Modeling

How shall we model the styles of composition, performance, and timbre for a better transfer effect? Most current studies use the following three approaches to model music styles: 1) to inherit the style implicitly from the training set (Hadjeres and Pachet 2016; Liang 2016; Huang et al. 2017; Xia 2016), 2) to use simple style embedding for generation (Mao, Shin, and Cottrell 2018), and 3) to apply style-related constraints for generation. In other words, they all require a manually-defined style representation or style label for generation.

As stated earlier, *style transfer calls for disentanglement of content and style*. It would make more sense to *learn* the disentanglement rather than crafting it by hand. In the field of deep generative modeling, learning disentanglement has already attracted a vast amount of attention (Thomas et al. 2017; Karimi et al. 2017; Larsson, Nilsson, and Kågebäck 2017; Kim and Mnih 2017). For image generation tasks, adversarial training has achieved disentanglement of latent factors and been applied within the generative adversarial network (Chen et al. 2016) and variational auto-encoder (VAE) (Mathieu et al. 2016) framework. A pioneering study has applied the VAE framework for algorithmic composition (Roberts, Engel, and Eck 2017). Though the convincing results are still bounded by the length of two bars, it is conceivable to apply it for style transfer task with some modification.

Upon a successful disentanglement, style can be considered as one of the latent factors and style transfer can be accomplished in two steps. The first is to disentangle a "style" code from the hidden representation that generates the music, and second is to "plug" such code into an appropriate sequence generation framework that preserves all other factors.

Conclusion

In conclusion, music style transfer is a new research field which promises novel computational tools to generate both creative and human-like music. Questions like "what if Miles Davis wrote *Twelve Variations on 'Ah vous dirai-je, Maman'* and performed it on a flute" would be no more purely imaginary. In order to generate meaningful results, future works should be aware of the multi-level, multi-modal music representation and be clear whether the focus is timbre style transfer, performance style transfer, or composition style transfer. Also, the automatic disentanglement of content and style representation is the key for high-quality style transfer algorithms and worth the effort from the whole field, and we believe that it is an efficient way, if not the only way, towards a complete end-to-end, cross-modal music style transfer system.

References

Ananthabhotla, I., and Paradiso, J. A. 2017. Visualsoundtrack: An approach to style transfer in the context of soundtrack prototyping. In *International Computer Music Conference (ICMC-2017)*.

Boulanger, R. C. 2000. *The Csound book: perspectives in software synthesis, sound design, signal processing, and programming*. MIT press.

Briot, J.-P.; Hadjeres, G.; and Pachet, F. 2017. Deep learning techniques for music generation-a survey. *arXiv preprint arXiv:1709.01620*.

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

Cope, D., and Mayer, M. J. 1996. *Experiments in musical intelligence*, volume 12. AR editions Madison, WI.

Dannenberg, R. B. 1993. Music representation issues, techniques, and systems. *Computer Music Journal* 17(3):20–30.

Dannenberg, R. B. 1997. Machine tongues xix: Nyquist, a language for composition and sound synthesis. *Computer Music Journal* 21(3):50–60.

Dmitry, U., and Vadim, L. 2016. Audio texture synthesis and style transfer. https://dmitryulyanov.github.io/audio-texture-synthesis-and-style-transfer.

Engel, J.; Resnick, C.; Roberts, A.; Dieleman, S.; Eck, D.; Simonyan, K.; and Norouzi, M. 2017. Neural audio synthesis of musical notes with wavenet autoencoders. *arXiv preprint arXiv:1704.01279*.

Gao, Y. 2017. Towards neural music style transfer. Master Thesis, New York University. https://github.com/821760408-sp/the-wavenet-pianist.

Gatys, L. A.; Ecker, A. S.; and Bethge, M. 2015. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*.

Hadjeres, G., and Pachet, F. 2016. Deepbach: a steerable model for bach chorales generation. *arXiv preprint arXiv:1612.01010*.

Hu, N. 2004. *Automatic Construction of Synthetic Musical Instruments and Performers*. Ph.D. Dissertation, Carnegie Mellon University.

Huang, C.-Z. A.; Cooijmans, T.; Roberts, A.; Courville, A.; and Eck, D. 2017. Counterpoint by convolution. In *18th International Society for Music Information Retrieval Conference (ISMIR-2017)*.

Kaliakatsos-Papakostas, M.; Queiroz, M.; Tsougras, C.; and Cambouropoulos, E. 2017. Conceptual blending of harmonic spaces for creative melodic harmonisation. *Journal of New Music Research* 46(4):305–328.

Karimi, A.-H.; Banijamali, E.; Wong, A. W.; and Ghodsi, A. 2017. Jade: Joint autoencoders for dis-entanglement. In *Learning Disentangled Representations, NIPS 2017 Workshop*.

Kim, H., and Mnih, A. 2017. Disentangling by factorising. In *Learning Disentangled Representations, NIPS 2017 Workshop*.

Kirke, A., and Miranda, E. R. 2009. A survey of computer systems for expressive music performance. *ACM Computing Surveys (CSUR)* 42(1):3.

Larsson, M.; Nilsson, A.; and Kågebäck, M. 2017. Disentangled representations for manipulation of sentiment in text. In *Learning Disentangled Representations, NIPS 2017 Workshop*.

Lattner, S.; Grachten, M.; and Widmer, G. 2016. Imposing higher-level structure in polyphonic music generation using convolutional restricted boltzmann machines and constraints. *arXiv preprint arXiv:1612.04742*.

Liang, F. 2016. Bachbot: Automatic composition in the style of bach chorales. Masters thesis, University of Cambridge.

Loy, G. 1989. Composing with computers: A survey of some compositional formalisms and music programming languages. In *Current directions in computer music research*, 291–396. MIT Press.

Malik, I., and Ek, C. H. 2017. Neural translation of musical style. *arXiv preprint arXiv:1708.03535*.

Mao, H. H.; Shin, T.; and Cottrell, G. W. 2018. Deepj: Style-specific music generation. *arXiv preprint arXiv:1801.00887*.

Mathieu, M. F.; Zhao, J. J.; Zhao, J.; Ramesh, A.; Sprechmann, P.; and LeCun, Y. 2016. Disentangling factors of variation in deep representation using adversarial training. In *Advances in Neural Information Processing Systems*, 5040–5048.

McCartney, J. 1996. Supercollider: a new real time synthesis language.

Müller, M.; Prätzlich, T.; Bohl, B.; and Veit, J. 2013. Freischutz digital: A multimodal scenario for informed music processing. In *Image Analysis for Multimedia Interactive Services (WIAMIS), 2013 14th International Workshop on*, 1–4. IEEE.

Pati, A. 2018. Neural style transfer for musical melodies. Music Informatics Group, Georgia Tech Center for Music Technology. https://ashispati.github.io//style-transfer/.

Ramirez, R.; Maestre, E.; and Serra, X. 2010. Automatic performer identification in commercial monophonic jazz performances. *Pattern Recognition Letters* 31(12):1514–1523.

Roberts, A.; Engel, J.; and Eck, D. 2017. Hierarchical variational autoencoders for music. In *31st Conference on Neural Information Processing Systems (NIPS 2017) Workshop*.

Russ, M. 2004. *Sound synthesis and sampling*. Taylor & Francis.

Schoenberg, A., and Stein, L. 1969. *Structural functions of harmony*. Number 478. WW Norton & Company.

Simon, I., and Oore, S. 2017. Performance rnn: Generating music with expressive timing and dynamics. https://magenta.tensorflow.org/performance-rnn.

Stamatatos, E., and Widmer, G. 2005. Automatic identification of music performers with learning ensembles. *Artificial Intelligence* 165(1):37–56.

Thomas, V.; Bengio, E.; Fedus, W.; Pondard, J.; Beaudoin, P.; Larochelle, H.; Pineau, J.; Precup, D.; and Bengio, Y. 2017. Disentangling the independently controllable factors of variation by interacting with the world. In *Learning Disentangled Representations, NIPS 2017 Workshop*.

Verma, P., and Smith, J. O. 2018. Neural style transfer for audio spectograms. *arXiv preprint arXiv:1801.01589*.

Wang, G., and Cook, P. R. 2003. Chuck: A concurrent, on-the-fly, audio programming language. In *International Computer Music Conference (ICMC-2003)*.

Widmer, G., and Goebl, W. 2004. Computational models of expressive music performance: The state of the art. *Journal of New Music Research* 33(3):203–216.

Wiggins, G. A.; Müllensiefen, D.; and Pearce, M. T. 2010. On the non-existence of music: Why music theory is a figment of the imagination. *Musicae Scientiae* 14(1_suppl):231–255.

Xia, G. 2016. *Expressive Collaborative Music Performance via Machine Learning*. Ph.D. Dissertation, Carnegie Mellon University.

Zalkow, F. 2016. *Musical Style Modification as an Optimization Problem*. Ann Arbor, MI: Michigan Publishing, University of Michigan Library.