

Transforming Music into Chinese Musical Style based on Nyquist

Xinying Lyu *

Department of Engineering Physics, University of Illinois Urbana-Champaign, Champaign, United States

* Corresponding author email: xinying7@illinois.edu

Abstract. As music recomposing becoming a popular way of artistic creation, computer-aided music making has generated many neoteric pieces through algorithms. Style transfer is one of the methods of recomposition based on an existing musical piece. This paper aims to develop a method that could implement music style transferring to Chinese style by changing the mode into a pentatonic scale. It includes two methods in note-to-note modification that both achieved the purpose well and are suited for different situations. The output of the approach is proved to be successful by certain comparison between the existing Chinese style music and the one that was modified by the method. It is convenient for musicians to utilize the analysis to improve the efficiency of recomposition based on style changing. In addition, they will be given a general view of what the changed music could be without excessive thinking prior to the start of their composing process. These results shed light on guiding further exploration of music composing based on Nyquist.

Keywords: Style Transfer; Mode Changing; Nyquist; Note Transposition.

1. Introduction

A musical style is a particular term that specifies a class of music that was characterized by a period of history where it was developed, a musician that creates it, a genre, or even a culture [1]. Through the process of music development in various countries in the world, music can now be classified into different genre and styles based on their tonality, rhythm, music modes and timbral elements. On this basis, the style of music can be changed to another existing one by note changing or rearranging. For instance, in previous research there exist style transferring accomplished by focusing on the case of accompaniment styles as a one-shot method [2]. One of the most typical features that differ one style from another is the musical mode of a piece. For example, the modern music usually uses heptatonic mode that includes 7 main notes in an octave, while some of the music only have 6 notes or even 5 main notes in an octave (e.g., Chinese music).

In ancient China, musical instruments were designed simply compared to most of the musical instruments that was invented in near centuries. Therefore, they mostly were not capable of playing various modes, which resulted in the wide usage of the pentatonic scale throughout the history of China. The five main notes in this pentatonic musical mode are respectively called “gong”, “shang”, “jue”, “zhi”, “yu” in Chinese. Musicians in modern society have concluded the corresponding notes C, D, E, G, A in national mode, or 0, 2, 4, 7, 9 in Nyquist steps [3]. To be specific, the Chinese pentatonic modes are only made up of major second notes and minor third notes, though there exists some hexatonic and heptatonic scales in different areas of China. Besides, there is evidence that the heptatonic scales with seven notes were becoming increasingly wide-used due to the impact of western music cultural-appreciation, which leads to more deviate notes (e.g., “BianGong”). It is the Western equivalent of note B, occurring in Chinese music, they were usually treated as an ornamental tone or passing tone. This approach mainly focuses on transferring to the five major notes that denotes the characteristic of ordinary Chinese pentatonic mode [4].

While style transfer or alter in musical pieces are usually considered as a time-consuming work to musicians, algorithm-based style transfer using coding method can be considered as a way to assist musical rearrangement. In the previous researches, various methods introduced by other researchers brings other perspectives of computer-aided style transfer of music, e.g., algorithms including Gan-

base models' generators are tested to be successful for image domain style transfer of music [5]. In this research, by combining the advantages of programming within Nyquist and utilizing the music synchronization functions and the unique data type "score", a mode transfer program is developed based on Nyquist. It changes the musical style to Chinese pentatonic mode from any input piano music pieces by changing the notes to match the pentatonic mode scale in a clear and ordered manner.

The methods in evaluating the efficacy of music style translation have not yet reached a consensus internationally. It is relatively simple and straightforward for listeners to determine whether the style transfer is success based on their music experiences. Nevertheless, since this evaluation is quite subjective, this process cannot easily be achieved using computer programs similar to an auto grader of an exam. Previous study discussed the capability of distinguishing music of different artists using a classifier model that classifies existing songs by different artist particularly by the singing component, and eventually been applied to the output of the program algorithms to judge their efficacy [6]. Inspired by the research, this study has come up to an idea of picking some of the existing well-known Chinese style music and input the midi file into the project, and compare the output music to the original music. Since the project aim to transfer music into Chinese pentatonic style, the output should not differ a lot as the input is already matching the target style. The rest part of the paper is organized as follows. In section 2, the data, software and methods used for this project will be introduced. In section 3, two models that both works for the transformation process will be compared in different aspects. Then in section 4, the quality of transformation output in pentatonic mode will be evaluated by certain methods introduced later.

2. Data & Method

2.1 Data

Musical Instrument Digital Interface (MIDI) is a representation of music that encodes the keyboard performances and continuous controls of the pitch, the velocity, the volume and the time of each event. This research utilizes the sequential behavior within Nyquist in order to transfer a midi file into a sequence that can be modified. The input midi file used for this study is the Minuet in G composed by Bach, a well-known classical piece of music that is simple but elegant to hear. The reason for choosing this as the music to transfer is that it has several ascending as well as descending scales which can test the consistency of the translated music, and chords are also involved in this piece which can test the whether the translated music is harmonious or not. The data type used in this project is the score, a graphical display of a set of notes that includes their timing, pitches, instruments, and dynamics. In Nyquist, it is a representation of lists of notes or sound events.

2.2 Model, Software & Evaluations

The program in this research was written within the NyquistIDE, which combines functions and interfaces that interact with sounds, music and waves. Nyquist is implemented in Java, interacts through sockets with SAL, and written in XLISP. The data structure "score" in Nyquist is a sequence of sound elements, as well as a quoted list, where the notes inside the list are in increasing order, with the attributes of time, stretch and behavior.

The evaluation process could be done by searching for some existing videos that has transferred a well-known musical piece to Chinese style successfully based on the amount of listeners in the comment that considered it as a success, record the notes that was altered. Then, one generates the transferred music through the project code, and compares the output to the one in the video. In addition, directly inputting Chinese music into the code would also determine the degree of appropriateness of the style transfer, as a successful Chinese Pentatonic transfer code would retain the original notes mostly at the same position as before.

2.3 Procedure

After learning the basic skills of Nyquist and Lisp through the learning materials provided by Professor Roger Dannenberg, an idea of creating a piece of computer music is came up using Nyquist functions that has one main theme but consist of three different musical styles, i.e., the first part could be classified as rock music, the second part could be jazz, and the third part could be Chinese music. What the algorithm could accomplish is adding some typical characteristics of that style. For example, rock music, the sound of drum set and electric guitar would turn a melody more closely towards the common rock songs. To better realization, it focuses on some more specific characteristics of the musical style that was chosen and make variations on that feature, so that the evaluation process would not be too broad to be considered. After listening to several musical pieces in different styles and searching for some of different music characteristics, it was discovered that the translation in music mode is a significant process in style transfer for music between different cultures, especially for Chinese music. Meanwhile, the data structure “score” within Nyquist is capable of generating or altering notes within a sequence. Therefore, the ultimate technique the group wanted to achieve has become changing existing musical score into different music modes that corresponds to music styles.

To create an algorithm, the scale of the Chinese pentatonic modes are investigated to familiarize it with implementation on musical pieces. Then, one considered about the possibilities of ways that could change a heptatonic mode into a pentatonic mode which has less notes per octave. Inspired by the research of Generalized Diatonic and Pentatonic Scales: A Group-Theoretic Approach, the notes that occur in the piece to be modified can be classified into groups within a 12-tone scale, and each note can be defined as the primitive elements plus some steps. On this basis, the process of note transposition can be limited within 12 changes in the coding [7]. Two methods were eventually concluded to be construct in Nyquist, which are discussed in details in the next part of this paper.

For the original music that was to be modified, the author first thought of importing the score note by note into a new timed-sequence created in Nyquist based on the score of that music found online. While this method does work and is able to generate the input score needed, it is quite time-consuming and tedious since there are hundreds of notes for longer music pieces. Therefore, this study utilized the midi extension package in Nyquist so that the score could be automatically generated by a few command lines. Besides, the output wave file would be automatically stored as a temporary file within the computer, which could be easily accessed through looking into that directory. A function that transfer the output to midi was also implemented in case of further changes to be made.

3. Results & Discussion

3.1 Score Implementation

In order to take advantage of the convenience of midi file usage in implementing the music score notes, Nyquist midi extension package was downloaded from the website which enables modification of midi file within Nyquist. The code for inputting midi file and transfer it into a datatype of score is presented in Table. 1.

Table 1. Code for importing the score

Step 1	Use seq-create() to set a sequence “my-seq”
Step 2	Load the midi directory
Step 3	Use open-binary() function to open the input midi file into a variable called “midi-file”
Step 4	Read in the notes into a variable called “midi-file” using the function seq-read-smf
Step 5	Close the midi-file
Step 6	Set a variable score to be the sequence from my-seq

To utilize this code, the input midi music file should be downloaded and saved within the midi directory in the Nyquist folder, and the only thing that is needed to be changed for various different input is the file name on line 3 within the parenthesis. This code would successfully convert the midi

file into a sequence of sound events of the score generated, each representing a note at different time intervals with a pitch.

3.2 Comparison of the Models

Two methods are adopted for translating music into Chinese pentatonic mode in order to determine the capability of each method in handling a typical music chosen. The first method could be called the “moving up” method. It needs the user to know the tonality of the music first, which could usually be determined by the ending note of the whole piece, and then this note is selected to be the fundamental note or base note, and further changes to the scale were made starting with it. Musical notes transposition can be done with finding the base note which is at step 0, and add a few steps each time to form a complete scale for this major [8]. For example, in the piece “Minuet in G”, the tonality is G major, then in order to translate the mode to pentatonic, all the notes in the octave that doesn’t match the pentatonic scale could be moved upwards a few steps to fit in the scale “G, A, B, D, E” respectively in order. Since the heptatonic scale has “G, A, B, C, D, E, F#” in one octave, C should be fit into D, D should be fit into E, E should be fit into the next G, etc. The accidentals were converted to main notes within the scale as well. Nyquist utilize steps to represent the pitches of notes, each step denotes an interval of a “flat” or “sharp” in music scales. By using the feature of frequency doubles every 12 steps, which is an octave, the author has applied a formula $note = (l - 7) \% 12$ to find the position of current note within the octave. “l” here is the note represented in steps, and 7 is the step of G, subtracting this 7 from the original step would tell the position of the note within the G major scale. The number 7 can also be substituted by other numbers depending on the fundamental note of the tonality. The coding were presented below in Table 2.

Table 2. Code for method 1

Step 1	Declare a function called add-accents (time, dur, expr).
Step 2	Set a variable <i>l</i> to be each note in the input file read-in.
Step 3	Set a variable <i>note</i> to be $(l - 7) \% 12$, the position of each pitch within each octave represented by steps.
Step 4	Using the function expr-set-attr(), If note = 1, increase the pitch by 1 step If note = 3, increase the pitch by 1 step If note = 5, increase the pitch by 2 step If note = 6, increase the pitch by 1 step If note = 7, increase the pitch by 2 step If note = 8, increase the pitch by 1 step If note = 9, increase the pitch by 3 step If note = 10, increase the pitch by 2 step If note = 11, increase the pitch by 3 step
Step 5	Return the list (time, dur, expr)

During the process of debugging, the input file is changed into another self-created score of a G major scale. Whether the code is able to change all the origin notes into a pentatonic sequence in different octaves will be tested. This “moving up” method creates a more melodious output music since there are no repeating notes, but the theme and notes has changed more to some extent than the original file. When one inputs a Chinese music to this algorithm, the main theme of the melody would possibly preserve less, especially when the range of notes in a row is wide.

For the second method, instead of shifting all the notes upwards, only the 5th note and the 11th note would be “leaned to adjacent” to fit the pentatonic scale. In this algorithm, the position of current note is still approached by the formula $note = (l - 7) \% 12$. The author lowered the pitch of the 5th note by 1 step, and increased the pitch of the 11th note by 1 step, so that no notes of C and F# would be present in this G major music piece. For accidentals, they were altered to match the nearist note in the pentatonic scale just like the previous method. The coding were presented in Table 3.

When one inputs a Chinese music to this algorithm, it would preserve most of the original notes within the score, since it is merely changing the 5th step, the 11th step and those notes with accidentals instead of shifting all the notes upwards like the first method. However, repeated notes are possibly occurring which slightly destructs the overall fluency of the music.

Table 3. Code for method 2.

Step 1	Declare a function called add-accents (time, dur, expr).
Step 2	Set a variable l to be each note in the input file read-in.
Step 3	Set a variable $note$ to be $(l - 7) \% 12$, the position of each pitch within each octave represented by steps.
Step 4	Using the function expr-set-attr(), If $note = 1$, decrease the pitch by 1 step If $note = 3$, decrease the pitch by 1 step If $note = 5$, decrease the pitch by 1 step If $note = 6$, increase the pitch by 1 step If $note = 8$, increase the pitch by 1 step If $note = 10$, increase the pitch by 2 step If $note = 11$, increase the pitch by 1 step
Step 5	Return the list (time, dur, expr)

3.3 Evaluations of the Method

The musical piece chosen for evaluation is called *The Cowherd's Flute*, a well-known Chinese piano piece composed by He, also called Mutongduandi in Chinese. To demonstrate the effectiveness of this program, the author decides to compare the input scale and output scale and see if there are countable amounts of differences in notes between these two. As the input piece is already a Chinese style music, it should mostly follow the pentatonic mode with a few deviancies that are negligible, which should result in the output scale nearly not changed at all. When inputting the midi file of *The Cowherd's Flute* to the algorithm with the “moving up” method, based on the way it shifts the notes, the output music does not sounds like the original piece at all. Due to its wide range of notes and the fact that this algorithm only increases the step of notes after step 4 for each octave, different from the result of the Minuet in G piece, more unexpected repeating notes were occurring, and some ascending scales were destructed with a slight descending present within this series of notes. When inputting a Chinese music to the “lean to adjacent” method, the results were surprisingly resounding, only one note within the first movement was changed, the rest of the notes were all preserved well compared to the original scale. The repeated notes problem which was considerable with Minuet in G was no longer an issue for this Chinese piece.

Nevertheless, for both methods, a drawback of this program is that it cannot successfully deal with tonality changes within the piece, since the number inside the formula correlates to the tonality was set up before inputting the file. For the piece *The Cowherd's Flute*, it starts with the C major, the formula of note should be merely $1 \% 12$, but when the music comes to the second movement, it changes to G major, which should have the formula switched into $(1 - 7) \% 12$.

3.4 Application

To sum up, this study has successfully created a way of transferring music styles using Nyquist functions as long as a midi input is provided. This program is applicable not only for the pentatonic style transfer, but also for all other modes that could utilize the similar algorithm by changing some numbers as long as the scale of that mode is given. Compared to the ordinary method of changing music styles by changing the duration and pitch of each note sequentially, though the score rearranged by musicians note by note might be more melodious and fluent, using this Nyquist project to generate a translated music increases the overall efficiency and flexibility of style transfer. Most related work that was done before were focusing on utilizing computer programs to extract information from music data, but none of them attempt to change the music style of an already existing music mode through

the computer program merely. It is convenient and time-saving for musicians or composers to make additional changes based on the output file. For future uses, it is possible to create synthesized music data sets with different style transferring through variational auto-encoders and Cycle GANs as mentioned in previous research [9].

4. Limitation & Prospects

As mentioned above, the limitation of this study is that it cannot handle longer musical pieces that has movements with different tonality, and at the same time, the user must be able to determine the major of the piece and make changes to the code in order to get the correct output file. However, considered about the environments would be used for this research are usually going to be for music creations instead of for those that know nothing about music, this limitation doesn't affect the overall functionality of the project much. Moreover, this study could only make translation to music pieces that only includes piano as a musical instrument. Nyquist doesn't have a well-prepared data type that could translate an orchestra ensemble into a Nyquist score note by note, which can be a future research orientation to be working on.

Since this study is a mode transferring technique for pentatonic scales, it can be further modified to fit other modes like the Blues mode or Dorian modes as long as the information of the musical scale of that mode is provided. Inevitably, the less notes are included in each octave for a certain mode, the less melodious the output would sound since the number of notes cannot be reduced randomly. Moreover, this program only changes the music mode technically, i.e., it merely makes variations on the notes of the original scale of music, while in reality, style transfer includes more changes on the prospects of timbre, rhythm, musical instruments, etc. Considered about the algorithm of the localized key finding method, the program could be improved by altering the selection of the base note to be the one that occurred the most using the recursive median filter, based on the tonal hierarchy and the fact that the most stable pitch classes should occur the most often [10].

5. Conclusion

In conclusion, this paper investigates the feasibility of transferring musical piece into Chinese pentatonic mode while knowing the tonality of this piece through Nyquist. Specifically, making note-to-note transposition so that all the notes within the piece to be transferred is matching the pentatonic scale. According to the analysis, the "moving up" method tends to return a more harmonic and melodious output and works better for pieces with smaller note span, while the "lean to adjacent" method restores the main theme of the original piece better and technically returns a more "Chinese styled" music, but sometimes creates some unwanted repeating notes. In the future, further algorithms can be added on to the existing algorithm to implement auto finding of the tonality and sectional type mode transferring for tonality changes within one piece. Overall, these results offer a guideline for more efficient music creation and variation.

References

- [1] Shan M K, Kuo F F. Music style mining and classification by melody. *IEICE TRANSACTIONS on Information and Systems*, 2003, 86(3): 655-659.
- [2] Cífka O, Şimşekli U, Richard G. Groove2Groove: one-shot music style transfer with supervision from synthetic data. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2020, 28: 2638-2650.
- [3] Chinese pentatonic scales. *SCRIBD*. Retrieved from: <https://www.scribd.com/doc/180658975/Chinese-Pentatonic-Scales>.
- [4] Lu-Ting H, Kuo-Huang H. On Chinese scales and national modes. *Asian music*, 1982, 14(1): 132-154.

- [5] Brunner G, Wang Y, Wattenhofer R, et al. Symbolic music genre transfer with cyclegan. 2018 ieee 30th international conference on tools with artificial intelligence (ictai). IEEE, 2018: 786-793.
- [6] Heller B, Ryzhik A, Tesfai Z. Evaluation of Vocal Audio Style Transfer. Retrieved from: [https:// github.com/ zarahatesfai/cs229project](https://github.com/zarahatesfai/cs229project).
- [7] Zweifel P F. Generalized diatonic and pentatonic scales: a group-theoretic approach. Perspectives of new music, 1996: 140-161.
- [8] Zhu Y, Kankanhalli M. Music scale modeling for melody matching. Proceedings of the eleventh ACM international conference on Multimedia. 2003: 359-362.
- [9] Cífka O, Şimşekli U, Richard G. Supervised symbolic music style translation using synthetic data. arXiv preprint arXiv:1907.02265, 2019.
- [10] Shmulevich I, Yli-Harja O. Localized key finding: Algorithms and applications. Music Perception, 2000, 17 (4): 531-544.